Australian Government
Department of Defence
Defence Science and
Technology Organisation

Operation Manual for Measurement and Discrete Layer Peeling of Fibre Bragg Grating Spectra

Chris Brooks and Claire Davis

Air Vehicles Division
Defence Science and Technology Organisation

DSTO-TN-0801

ABSTRACT (U)

This document is a manual detailing the operation of an interrogation system for measuring the complex reflection spectrum of a Bragg grating. The manual also describes custom-designed software for deconvolving this data to determine the pitch profile of the grating by using a discrete layer peeling technique. This information can be used to determine the strain profile experienced by the grating in cases where the grating length spans the localised strain field.

RELEASE LIMITATION

Approved for public release

Published by

Air Vehicles Division
DSTO Defence Science and Technology Organisation
506 Lorimer St
Fishermans Bend, Victoria 3207 Australia

Telephone: (03) 9626 7000
Fax: (03) 9626 7999

AR-014-092
February 2008

Operation Manual for Measurement and Discrete Layer Peeling of Fibre Bragg Grating Spectra

Executive Summary

Conventionally, strain sensing on structures is achieved via electrical resistance foil gauges which are used as point sensors or arranged into strips to form arrays of strain sensors. A single optical fibre can contain an array of Bragg gratings at different wavelengths and has the potential to replace arrays of electrical gauges. However, used in this manner, Bragg gratings can still only provide a distributed strain profile with a spatial resolution limited by the grating density and gauge length.

In theory, the spectrum of a grating which has a gauge length longer than the strain field of interest can be deconvolved to offer a spatially continuous measurement of the strain profile. The deconvolution method requires both the intensity and phase spectrum from the Bragg grating to reconstruct the refractive index pitch variation (and hence the strain profile) using a discrete layer-peeling technique.

This document is a manual detailing the operation of an interrogation system for measuring the complex reflection spectrum of a Bragg grating. The manual also describes custom-designed software for deconvolving this data to determine the pitch profile of the grating by using a discrete layer-peeling technique. This information can be used to determine the strain profile experienced by the grating in cases where the grating length spans the localised strain field.

Contents

1. Hardware Setup

1.1 Connecting the Hardware

The Distributed Strain Measurement System (SMS) is designed to work with an Agilent (HP) 8164A Mainframe and an optional JDS SB series fibre optic switch. Both of these devices are connected using a General Purpose Interface Bus (GPIB). These devices must be connected to the computer's GPIB port before starting the SMS software. Multiple devices can be connected to a single GPIB port by using double-sided connector cables.

Most computers do not have a built-in capacity for GPIB connections so an interface card or adapter is usually required. Before running this program, install the interface card or adapter, if required, and its associated drivers.

Offline Mode

The SMS can be run without connecting to any hardware. No scans can be performed while in this "offline" mode, however, all the analytical features are fully functional and previously saved scan spectra can be processed. The mode is activated by clicking the "offline" button in the settings window.

1.2 Optical Modules

The Agilent 8164A mainframe can be loaded with several optical modules. The SMS requires that there be a tuneable laser module and at least two optical power detectors attached. The two power detectors can comprise either a single dual-detector module or two single-detector modules.

To simultaneously make reflectivity and interference measurements three optical power detectors are required. If there are only two detectors available an optical switch can be used to automatically change between reflectivity and interference measurement modes. If an optical switch is not used then the fibre connections must be manually changed in-between reflection and interference measurements.

At present, the system consists of an 81682A tuneable laser, an 81635 dual detector module and a series fibre optic switch. Table 1.2.1 lists the other modules supported by the software.

Table 1.2.1: Hardware modules supported by SMS

Laser Modules	Power Meter Modules
81680A	81630B
81682A	81634B
81640A	81636B
81642A	81637B
81689A	81635A (Dual)

1.3 Optical Connections

In order to calculate the strain gradient along the length of a fibre Bragg grating the system must first measure its reflectivity and interference spectra. In order to take these measurements a 2x2 fibre coupler and a fibre reflector are required in addition to the grating and connecting fibres. An additional 2x2 coupler is also required if a third optical detector is used. The setup required for the different hardware configurations are shown in Figures 1.3.1, 1.3.2 and 1.3.3.

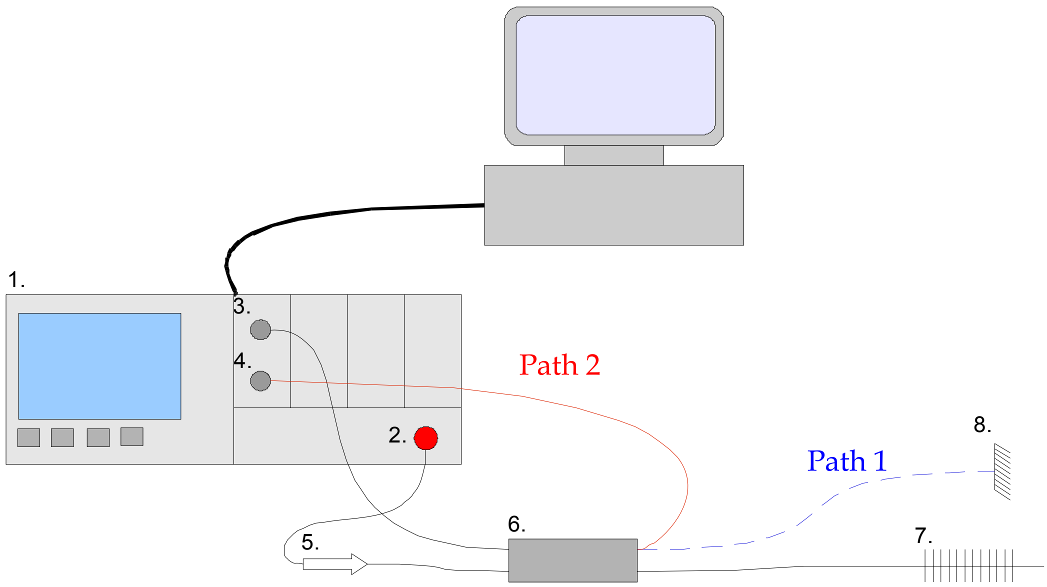


Figure 1.3.1: Hardware configuration with 2 detectors and no switch (Interference spectra recorded using path 1 and reflectance spectra recorded with path 2)

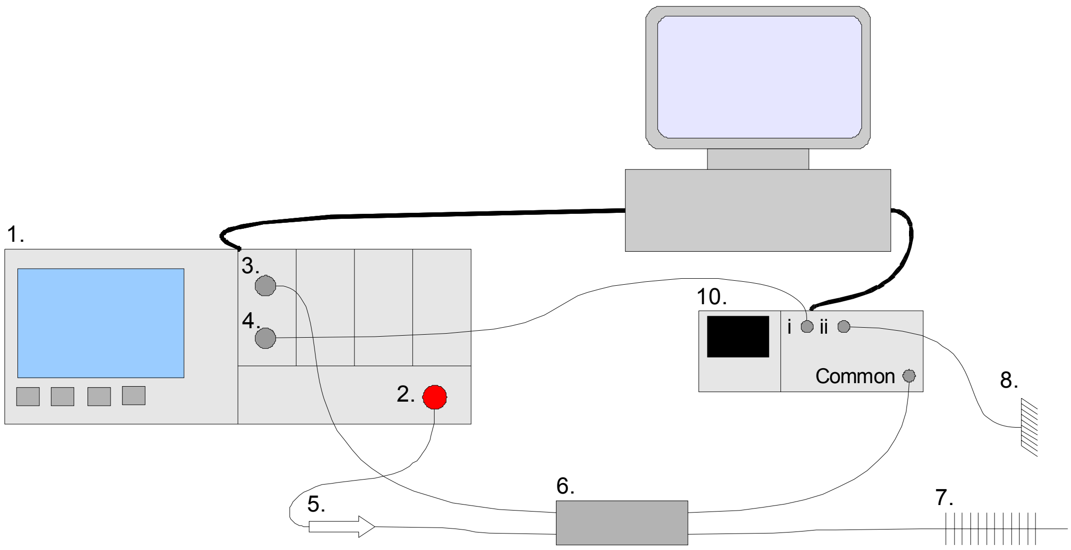


Figure 1.3.2: Hardware configuration with 2 detectors and a fibre optic switch (current configuration)

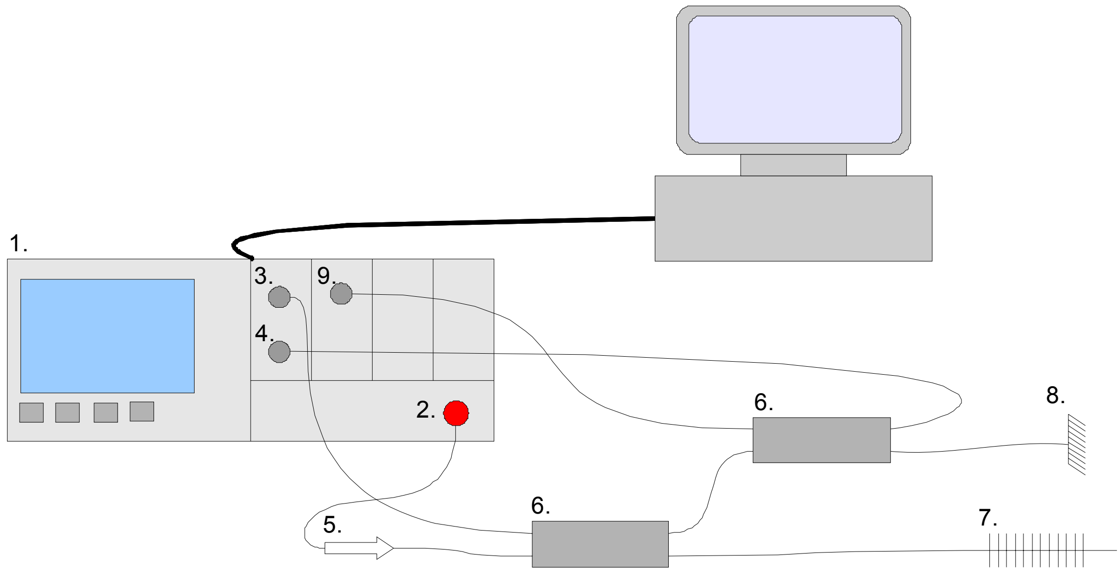


Figure 1.3.3:Hardware configuration with 3 detectors (no fibre optic switch required)

Table 1.3.1: Key to Components and Connections in Figures 1.3.1, 1.3.2 and 1.3.3

Item	Description	Default Address
1.	HP 8164A mainframe	GPIB:20
2.	Tunable laser output	Module:0
3.	Optical detector #1	Module:1 Channel:0
4.	Optical detector #2	Module:1 Channel:1
5.	Optical isolator	
6.	2x2 Fibre coupler (50:50)	
7.	Bragg grating specimen	
8.	Fibre-connected reflector or gold tipped fibre	
9.	Optical detector #3	Module:2 Channel:0
10.	JDS SB series optical switch	GPIB:10

Connection	Description
Thin	Optical fibre
Thick	GPIB cable
Red (Solid)	Reflectivity measurements
Blue (Dashed)	Interference measurements

1.4 Configuring the Software

Once the hardware is set up the SMS software can be started. If the hardware is not set up using the GPIB addresses and module numbers specified in Table 1.3.1 the software must be manually configured. If the software does not automatically bring up the settings window when initially run it can be accessed using the *Settings* button in the lower right of the main window.

The first step to manually configuring the software is to enter the GPIB Bus and Address of the devices. A GPIB bus refers to the interface used to connect to the computer. If the computer only has one interface then it will be numbered 0. The GPIB address is used to identify a device and is generally set on the device being connected.

To find this number on the HP 8164A press the *Config* button on the device and select *GPIB Address* from the menu. To find the GPIB address of a JDS SB series switch, simply press the GPIB button and the address will be displayed.

The software can attempt to automatically find the GPIB addresses of the HP 8164A and a JDS SB series switch. To attempt to automatically find the address press the *Search* button and enter the GPIB bus to search, which will be 0 for most cases.

Once the device addresses have been set the module numbers of the devices in the HP 8164A must be set. If a dual power detector module is used then the channel parameter can be used to set which detector is used. This setting should be 0 for a single power detector module. If a third power detector is to be used then the *Interference Power Sensor* check box should be selected. If an optical switch is being used then the output paths corresponding to a reflectivity and interference measurement should be entered.

Once these values are entered press the *Save Values* button to confirm the options selected. If there are no errors then the button will grey out and the settings window can be closed by pressing the *Done* button.

If the software could not communicate with the appropriate device or it doesn't appear to be the correct type, e.g. the module number entered for an optical detector is actually an optical source, then it will be highlighted in red. While there are errors present the settings can be saved by selecting the *Use Anyway* option when the pop-up appears detailing the error. However, in this case, the application will not run with certain logical errors, such as assigning one optical detector to measure two quantities.

2. Performing Scans

Once the hardware is connected and the software has been properly configured the system can be used to take measurements from a Bragg grating. There are only four parameters required to perform a scan. The main parameters are the start and end wavelengths. These should be set so the entirety of the grating's reflection spectrum will be measured.

Due to limitations in the hardware increasing the wavelength range may result in a reduction in resolution. The scan speed used sets the lower limit on the resolution as the device is only capable of sampling at 10kHz and storing a maximum of 16001 data points. The minimum resolution available at each scan speed is shown in Table 2.1. Depending on the wavelength range requested the scan resolution may be increased.

Table 2.1: Minimum wavelength resolution at a given scan speed

Scan Speed	Minimum Resolution
0.5 nm/s	0.001 nm
5.0 nm/s	0.005 nm
40 nm/s	0.040 nm

In practice the magnitude and profile of the strain applied to the grating is generally not known in advance. It may be appropriate to use a large wavelength range initially to determine the wavelength limits of the grating spectra and then reduce the wavelength range to maximize the resolution.

Any change in the strain profile will most likely change the wavelength range of the reflection spectrum. To avoid having to continually change the wavelength range as the strain changes the lower wavelength should be found while the grating is unstrained and the upper wavelength should be found while the grating is under the highest expected amount of strain.

The type of scan should then be selected using the *Scan Type* selector. Both a reflectivity and interference scan must be performed. If either an optical switch or third optical power detector is available then a complete scan may be performed. A complete scan will record both the reflectivity and interference spectra.

The power of the laser is set on the device itself. The software sets the power meters to read up to a maximum of 10dBm (10mW). The output of the laser should be set close to this level, without exceeding it.

Once all parameters are set the scanning process can be started by pressing the *Begin Sweep* button. **This will turn the laser ON if it is not locked.** If the laser is locked then a window will pop-up allowing the laser to be unlocked if desired. To ensure there is no inadvertent beam exposure, the laser should only be unlocked once all the fibre connections have been made.

3. Spectrum Processing

Before the reflectivity and interference spectra can be used to calculate the strain in the fibre some processing is required. This processing is performed in the *Reflectivity* and *Interference* windows selected using the tabs at the bottom of the main window. The *Phase* tab can be used to view the phase spectrum calculated from the interference fringes.

3.1 Reflectivity Spectrum

The Reflectivity section allows the offset of the reflectivity spectrum to be changed. These adjustments can be used to account for noise and back reflection in the system. The value of *Baseline* is added to the spectrum and in most cases will be negative to achieve a zero baseline. The reflectivity values will always be constrained between 0 and 1. This can result in clipping if the baseline is set too low. Clipping is undesirable and can cause incorrect strain calculations.

However, having many non-zero values outside of the grating reflection zone can also cause large amounts of noise in the grating inversion. This non-zero noise can be countered to some extent by setting the wavelength boundaries described in the proceeding paragraph. Some experimentation may be required to determine the optimal baseline value. Figure 1.3.1 shows the baseline of a reflectivity spectrum. Ideally this baseline should be set to 0, in this case by assigning the *Baseline* setting to -0.1.

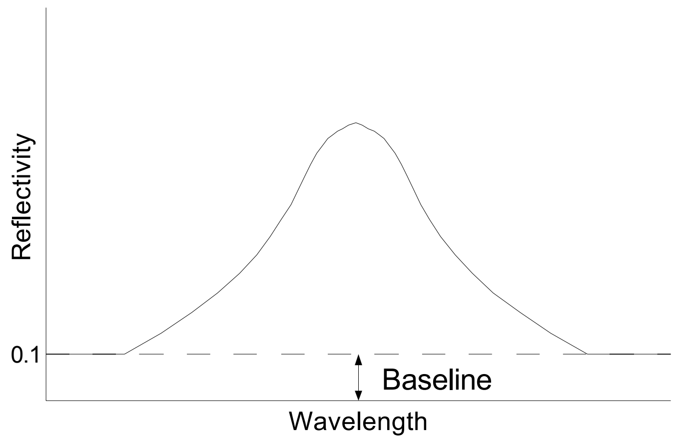


Figure 3.1.1: The baseline of a reflection spectrum

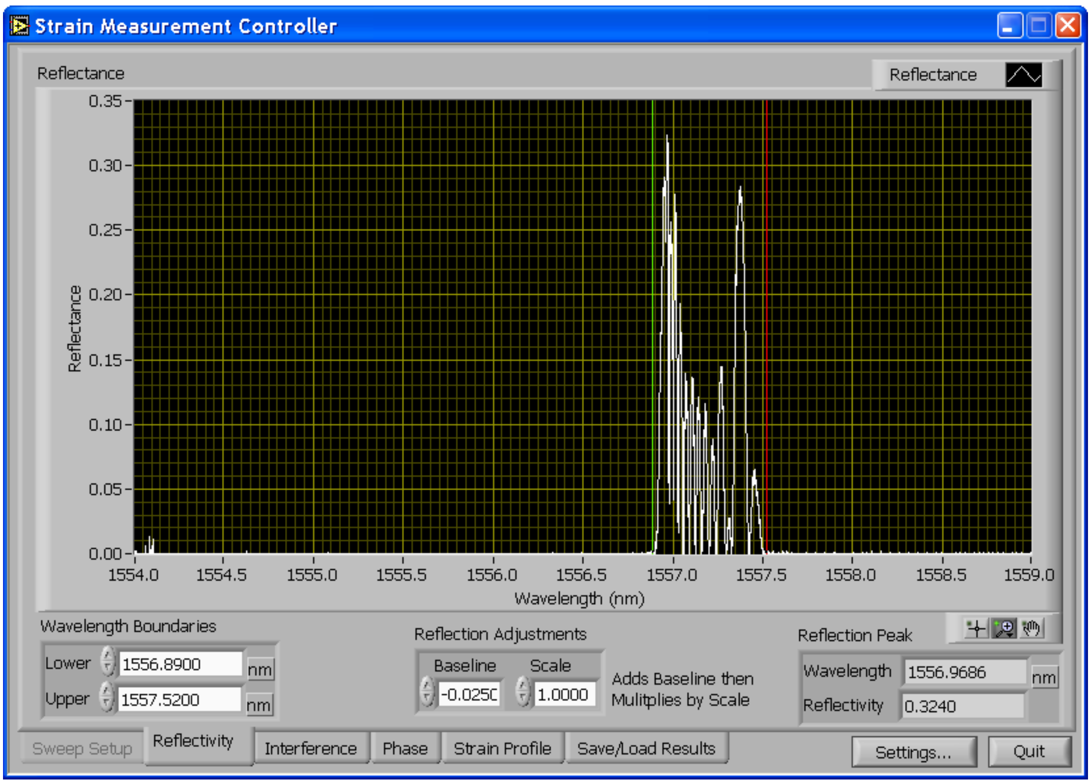


Figure 3.1.2: The reflectivity spectrum after processing

In order to only include the reflection spectrum of the grating, excluding any noise in the baseline, a lower and upper wavelength can be set. If these values are set then only wavelengths falling between these two boundaries will be used in strain calculations. The lower boundary is marked with a green line and the upper boundary marked with a red line, as can be seen in Figure 3.1.2.

3.2 Interference Spectrum

The interference spectrum is converted into a phase spectrum by fringe counting. Each fringe maximum is equal to an increase of 2π radians in phase. There are three different methods of performing this conversion, two automatic and one manual.

3.2.1 Simple Peak Detection

The *Simple Peak Detection* algorithm detects the peaks in the interference spectrum. The threshold value is used to prevent noise 'ripples' in the interference spectrum being detected as peaks. For example a threshold of 0.1 will remove all peaks which have a 'height' less than this value. The peak's height is defined as the difference in value between the peak and its closest minima as shown in the figure below.

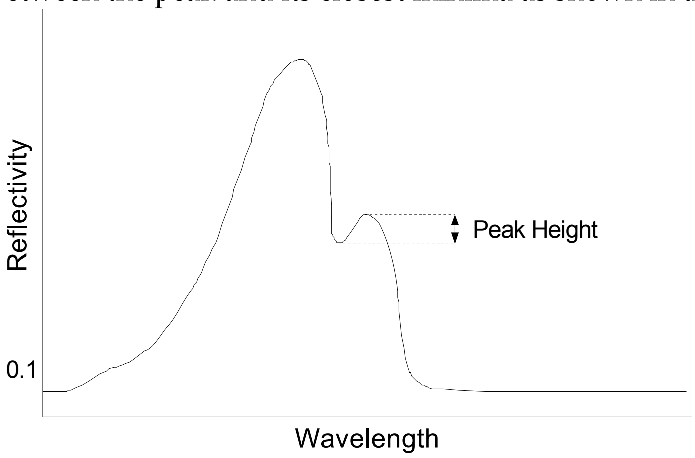


Figure 3.2.1: The peak height used for threshold comparison

3.2.2 Normalised Peak Detection

The *Normalised Peak Detection* divides the interference spectrum by the reflectivity spectrum to try and account for interference peaks in the lower reflectivity regions of the reflection spectrum, which might otherwise be below the threshold value. The *Offset* setting is used to adjust the baseline of the normalised interference spectrum and the *Threshold* setting is used to remove small peaks.

By changing to the *Detection Graph* tab, located at the top of the graph, the effects of these settings can be seen. If a threshold value is set then two horizontal green lines will be shown. Any peaks which occur within these lines will not be included. The offset should be adjusted so the central line sits in the centre of the normalised interference spectrum as shown in Figure 3.2.2. The peak detection graph is only useful when *Normalised Peak Detection* is selected.

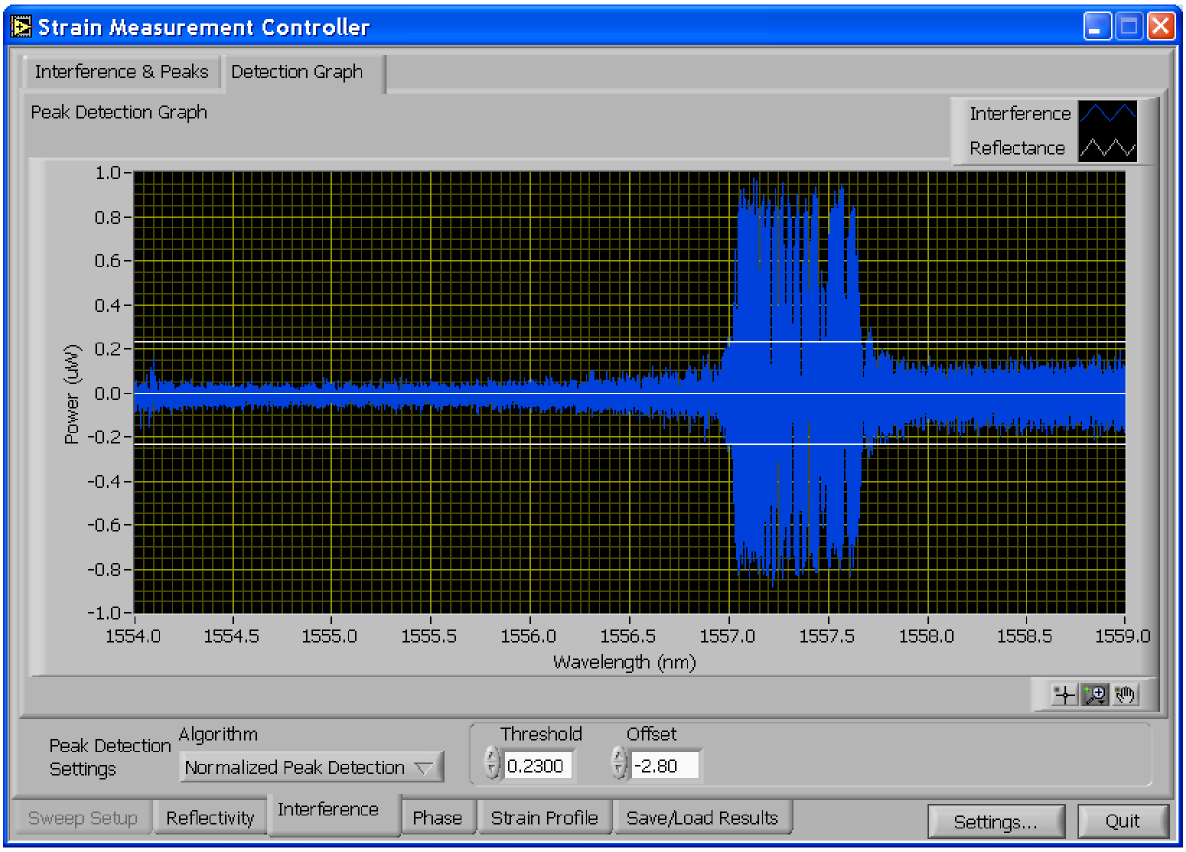


Figure 3.2.2: A correctly adjusted Peak Detection graph

3.2.3 Manual Peak Detection

The peaks can also be selected manually. Peaks can be selected and deselected by using the *Edit Mode* control. While in manual selection mode all points on the interference spectrum will be marked with boxes. A click on these boxes in either add or remove mode will either mark or unmark a point as a peak.

All selected peaks can be cleared by using the *Clear* button. The selected peaks can be saved and loaded later in the *Save/Load Results* tab.

Manual peak detection should only be used when the automatic methods are not performing correctly due to the time required to select each peak. The most noticeable indicator of the automatic methods failing is a large number of missing peaks in the strong reflectivity region of the spectrum.

Once the peaks have been selected, either automatically or manually, the resulting phase spectrum can be seen in the *Phase* tab, where it is overlaid against the reflectivity spectrum.

4. Calculating the Strain

Once the reflection and interference spectra have been recorded and the resultant phase spectrum has been recovered the strain along the fibre can be calculated. The strain calculations are setup and performed in a separate window accessed by selecting the *Strain Profile* tab at the bottom of the window and clicking the *Calculate Strain* button.

4.1 Spectrum Pre-Processing

Before the grating spectra are inverted to produce a strain profile they may need to first undergo some pre-processing. Any changes made to the settings will be reflected in the graph above, which displays the reflectivity and phase of the reflection spectrum. There are three different pre-processing steps which are described below. In most cases there is no best value and some experimentation with the settings may be required to obtain the best strain profile.

Resolution

In order to calculate the strain profile the data points in the reflection spectrum need to be evenly spaced. When the reflection spectrum is measured the points are not always spaced exactly evenly. This is due to slight variations in the trigger which begins the recording process. Generally the spacing difference only varies by a very small amount. The spacing of the points can be set using the *Input Data Point Spacing* setting.

There is no real advantage in reducing the point spacing below what the data was scanned at since this will not improve the results and increasing the number of points increases the calculation times.

Wavelength Range Extension

The resolution of the strain profile is dependent on the wavelength range of the input data. In order to improve the strain resolution and possibly improve the un-wrapping of the inverted strain profile the wavelength range can be increased. The range can either be set as a start and end wavelength or as a wavelength range, taken from the reflection peak. Increasing the wavelength range will cause the calculation to take longer to complete.
The extensions to the reflection spectrum can either be taken from the last value before the extension or set to 0.

Phase Interpolation

The phase spectrum is constructed from interference peaks and will have a much lower resolution than the reflection spectrum. The calculations performed to obtain the strain profile require that both the intensity and phase have the same number of data points. This requires interpolating the phase spectrum. There are two interpolation methods available; Linear and Spline.

Linear interpolation is a very simple algorithm however it can produce sharp changes in phase. Spline interpolation produces more continuous interpolations, however it does not always work with all data sets, in which case the application will fall back to linear interpolation and a message will be displayed indicating this.

4.2 Inversion Calculation

Once the spectra pre-processing is completed the inversion parameters can be defined by clicking on the *Inversion Calculation* tab. There are several parameters which need to be entered before the inversion can be performed and the strain calculated. These parameters are constant for each fibre specimen.

Clicking the *Fibre Parameters* button will open a small window where some physical parameters of the fibre and grating can be entered. The parameters should be self explanatory. The *Unstrained Peak Wavelength* is the wavelength of the peak reflection intensity when the grating is unstrained.

The *Grating Length* parameter specifies the physical length of the unstrained Bragg grating. The *Grating Start* parameter is required because the grating does not always start at position 0 when inverted. The process of finding the grating start is described in detail in Chapter 7.

Once all the parameters are entered the inversion processes can be started by pressing the *Invert Grating* button. Once the process is completed the resulting coupling coefficient will be displayed in the graph above. Next to the *Invert Grating* button the inversion residue is displayed. A smaller number here indicates a better inversion. Most inversions have a residue below 0.001 although the number will vary, especially if the spectrum is noisy. A large residue may indicate that the inversion process could not be properly performed.

4.3 Strain Post-Processing

Once the reflectance spectrum has been inverted the strain profile can be calculated. The final strain profile can be adjusted by altering the phase un-wrapping method and the averaging. Once the post-processing is completed pressing the *OK* button will return to the main interface.

Phase Unwrapping

The strain profile is obtained by taking the derivative of the argument of the complex coupling coefficient. However, the argument is constrained between $\pm 2\pi$. In order to take the derivative of the argument it must be unwrapped, this can be done in two different ways.

Forcing the strain to either positive or negative values can be done if this is known to be the case. The other method involves using a threshold to detect when the phase has wrapped around. The threshold can be adjusted to try and achieve the best results.

When the phase unwrapping is changed it is reflected in the complex coupling coefficient graph on the *Inversion Calculation* tab. Ideally the argument should be either increasing or decreasing smoothly. Sharp changes will result in large spikes in the strain profile.

Figures 4.3.1 and 4.3.2 show the phase profile (in red) of the same grating inversion with two different unwrapping thresholds. Figure 4.3.1 is set to a low threshold of 0.2π while Figure 4.3.2 uses a threshold of π, producing a better result.

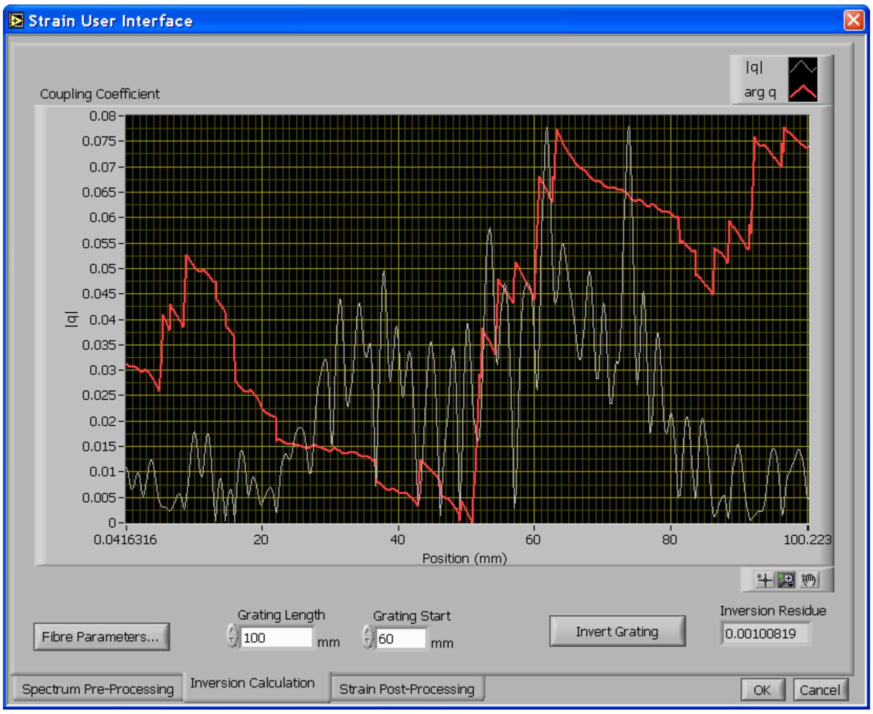


Figure 4.3.1: Coupling Coefficient graph with phase unwrapping using a threshold of 0.2π

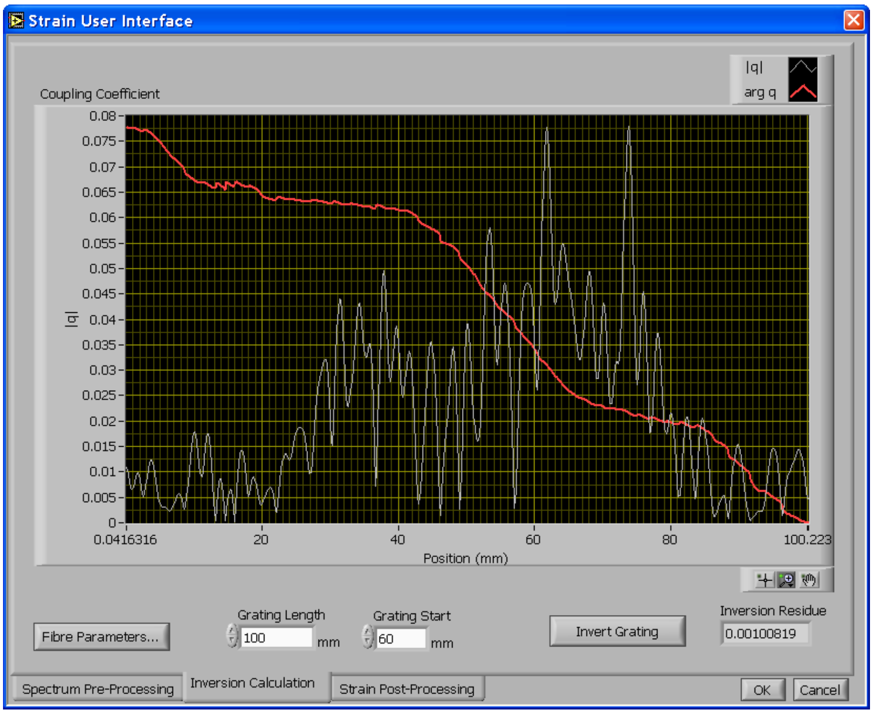


Figure 4.3.2: As in Figure 4.3.1 but using a threshold of π

Averaging
The calculated strain profile often has large spikes and dips caused by sharp changes in the coupling coefficient phase. The effect of these can be reduced by neighbourhood averaging. The averaging window is entered as either an absolute number of points or a relative percentage of the total points.

5. Saving & Loading Data

5.1 Saving Data

Both the reflection and interference spectra recorded and the phase spectrum and strain profile calculated can be saved for later reference or to be imported into another application. To save data the spectra and profiles required should be selected above the *Save* button. The following data can be saved: Reflection Spectrum, Interference Spectrum, Phase Spectrum and Strain Profile. Only the actual spectrum and profile data is saved. None of the settings used in calculating or modifying any profiles or spectra are saved and must be recorded separately if required.

Once the selections are made the data can be saved by pressing the *Save* button and entering the file name. The data will be saved in a tab separated text format which can be imported into most data analysis programs.

5.2 Loading Data

Previously saved reflection, interference and phase spectra can be loaded into the system. The indicators below the *Load* button show which spectra were found and loaded. Once loaded these spectra are treated in exactly the same way as if they were produced by a scan. The only exception is the phase spectrum which will be lost if any modifications are made to the peak detection settings.

The phase can optionally be automatically recalculated from the reflection and interference spectra using the current peak detection settings. If no phase spectrum could be found in the file it will be calculated from the reflection and interference spectra.

When a file is opened all modifications and settings, such as reflection adjustments, are reset to their default values.

5.3 Manually Selected Peaks

If manual peak detection is used then it is possible to save and load the selected peaks using the *Save* and *Load* buttons located at the top-right of the window. Note that the software only saves the numerical index of the point so the manual peaks file should be named in a way that links it with its associated spectra.

6. Measuring a Strain Profile, Step-by-Step

This section is designed to give a simple step-by-step guide to the common tasks required to take a strain profile measurement from a grating. Items marked with a * need to be performed for each scan, those marked with a ◊ usually only need to be entered once, but should be checked for each scan. All other items only need to be performed once unless the grating specimen is changed, in which case all steps will need to be repeated.

Recording the Spectra

1. Determine the required wavelength range from existing data, a long-ranged scan or an optical spectrum analyser. As the strain increases the wavelength range of the reflection spectrum will increase. Therefore this step may need to be repeated if a large increase in strain is applied.
2. Perform scan(s) for both Reflectivity and Interference spectra ensuring the entire grating reflection spectrum is recorded.*

Reflectivity Spectrum Adjustments

1. Set the *Lower Wavelength Boundary* value to just below the lower edge of the grating reflection spectrum to exclude any noise spikes in the lower wavelengths.◊ (See Figure 3.1.2)
2. Set the *Upper Wavelength Boundary* value to just above the upper edge of the grating reflection spectrum to exclude any noise spikes in the upper wavelengths.◊ (See Figure 3.1.2)
3. Adjust the reflection *Baseline* so the regions away from the grating's reflection spectrum are at or very close to zero with out clipping the grating reflection.◊
4. Adjust the reflection *Scale* to account for losses in the system.

Interference Peak Detection

1. Select the peak detection algorithm to use. In most cases the *Normalised Peak Detection* algorithm will work and should be tried first. Manual peak detection should only be used for either a small number of peaks or if the automated algorithms are unable to correctly detect the peaks.◊
2. If using an automated algorithm, adjust the algorithm parameters. Increasing the threshold above 0 will prevent small peaks from being detected. When using the *Normalised Peak Detection* algorithm the offset should be adjusted in the *Detection Graph* tab and should be aligned so the central white horizontal line is aligned with the baseline of the normalised interference spectrum, see Figure 3.2.2.◊
3. If using manual peak selection select the peaks.*
4. Check the generated phase profile. It there are any sharp changes in the region where the grating is reflecting then the detection parameters may need altering. The phase outside the grating's reflection range has little effect on the strain calculation and can be ignored.

Strain Calculation

Pre-Processing

1. Set the *Input Data Point Spacing* as desired.*
2. In the Wavelength Range Extension tab set the Wavelength Range to a larger value, since most of the spectrum outside the grating reflection was removed by the limiting boundaries. The optimal value will change depending on the grating, although in most cases a range of a least 5 nm is desirable.*
3. Set the *Phase Interpolation* to the desired value.*

Grating Inversion

1. Set the physical parameters of the fibre by clicking the *Fibre Parameters* button.
2. Find the start position of the grating as outlined in Chapter 7.
3. Set the length of the grating.
4. Perform the inversion by clicking the *Invert Grating* button.*

Post-Processing

1. Adjust the *Threshold* or *Phase Unwrapping Type* if needed to improve the strain profile.◊
2. Adjust the averaging if desired to attempt to reduce the number of sharp spikes in the strain profile.◊

Saving Data

1. Select the data sets to save and enter file name.

7. Finding the Grating Start

Finding the grating start can be a time consuming step requiring careful analysis of the grating inversion. Fortunately the grating start appears to be constant for each grating and only needs to be found once.

The grating start is found by performing a grating inversion but setting the grating start to 0 mm and setting the grating length to be much larger than it actually is. The extra length could initially start at 100 mm or 200 mm but using a larger value will cause the inversion to take a long time to complete. Once the *Invert Grating* button is pressed and the calculations are complete the coupling coefficient is displayed in the graph above. The main quantity of interest in finding the grating start is the absolute value of the coupling coefficient, q.

Once this 'extra long' grating has been inverted the absolute value of the coupling coefficient should be inspected to look for signs of the grating. The main signs are detailed below. If none of these signs are present then the start position may be further along the fibre and the *Fibre Length* setting should be further increased.

7.1 Grating Envelope

This is the quickest and easiest way to identify a grating and assuming that the noise in the reflectance spectrum is not too great this method should work.

Ideally $|q|$ should resemble the apodisation profile the grating was written with. Noise in the readings will distort this when inverted, although the general shape should remain. Most gratings are written with a Gaussian-like, or bell-shaped, apodisation profile. By examining the $|q|$ profile for a section resembling this shape with a length similar to that of the grating, the starting position can be found.

Figure 7.1.1 shows the refractive index profile and coupling coefficient profile, obtained from an inversion, of a 100 mm long Bragg grating. All inversions start from the point z = 0. If this grating was inverted with a length of 100 mm and a start of 0 mm then only the coupling coefficient profile in the region of 0 mm to 100 mm would be returned. From this information there is no way to determine the grating start position. However, if the grating length is set to 200 mm, for the purpose of determining the grating start position, then enough information would be returned from the grating inversion to allow the grating start to be located from the coupling coefficient profile.

Figure 7.1.2 shows a 100 mm grating inverted with a length of 200 mm and a start of 0 mm. Starting at ~60 mm a 100 mm bell-shaped curve fits closely to the $|q|$ graph, indicating a likely starting point of the grating of about 60mm.

If the graph appears flat, ignoring the noise and fluctuations, then the length may need to be increased to reach the grating.

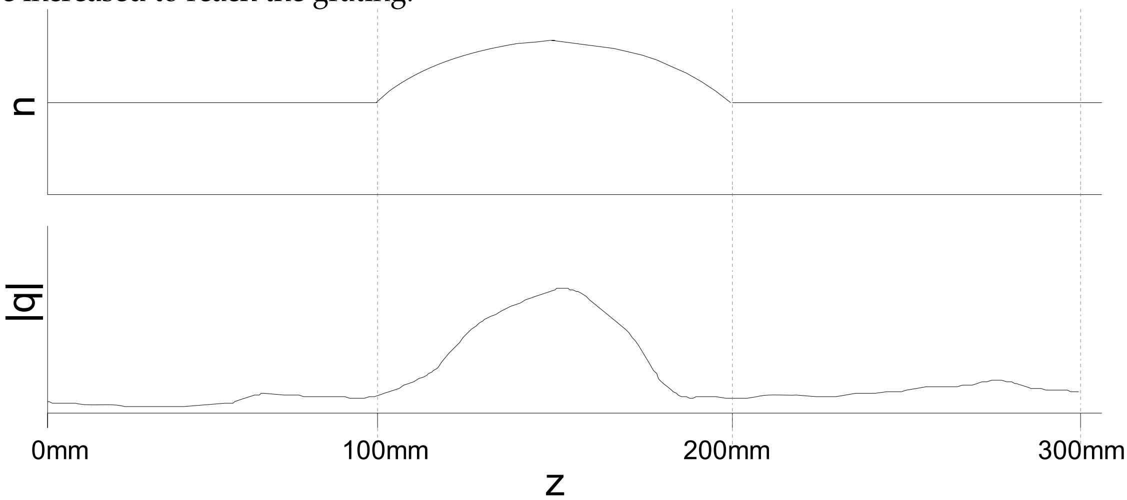


Figure 7.1.1: Refractive Index Profile and Inverted Coupling Coefficient Profile of a 100 mm Bragg Grating

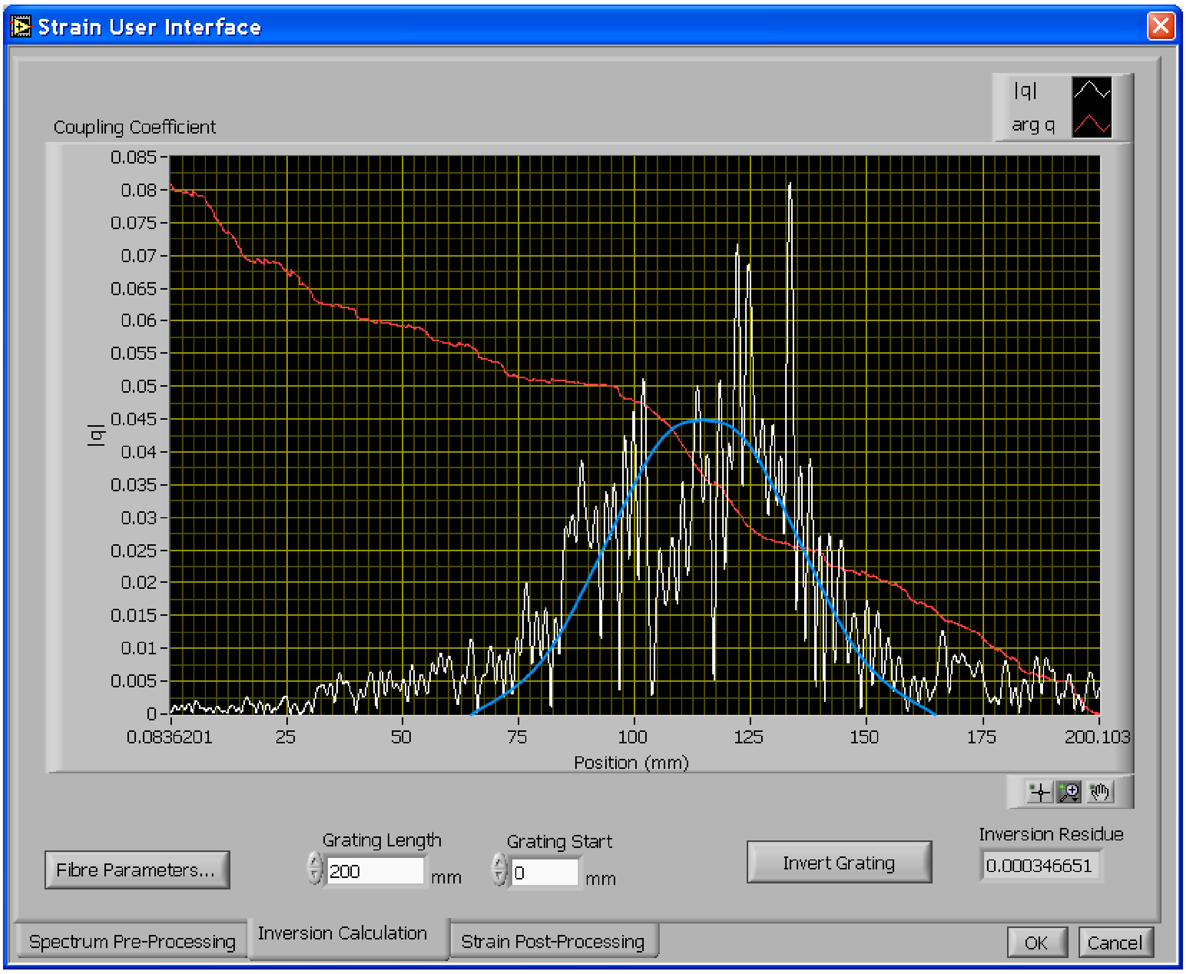


Figure 7.1.2: An extended inversion of a reflectance spectra to attempt to find the grating start position

7.2 Known Strain Profile

If the grating start position cannot be found by using the $|q|$ graph due to excessive noise then placing the grating under a known strain profile can be used to find the start position. Ideally the strain profile should have a distinguishing feature, such as a peak, which will show up on the strain profile. By locating this peak in the strain profile, and knowing its location relative to the grating, it is possible to approximately locate the grating start.

The biggest problem with this method is that the starting position can only be located to the accuracy of which the position of the grating in the fibre is known. Without knowing the exact distance of the grating start from the strain feature this method cannot exactly determine the grating start. However, it can provide a good approximation.

Figure 7.2.1 shows the same 100 mm grating inverted with a length of 200 mm and a start of 0 mm. The strain peak is known to be approximately 50 mm from the start of the grating. As the strain peak is around 110mm, this would place the starting position at approximately 60 mm.

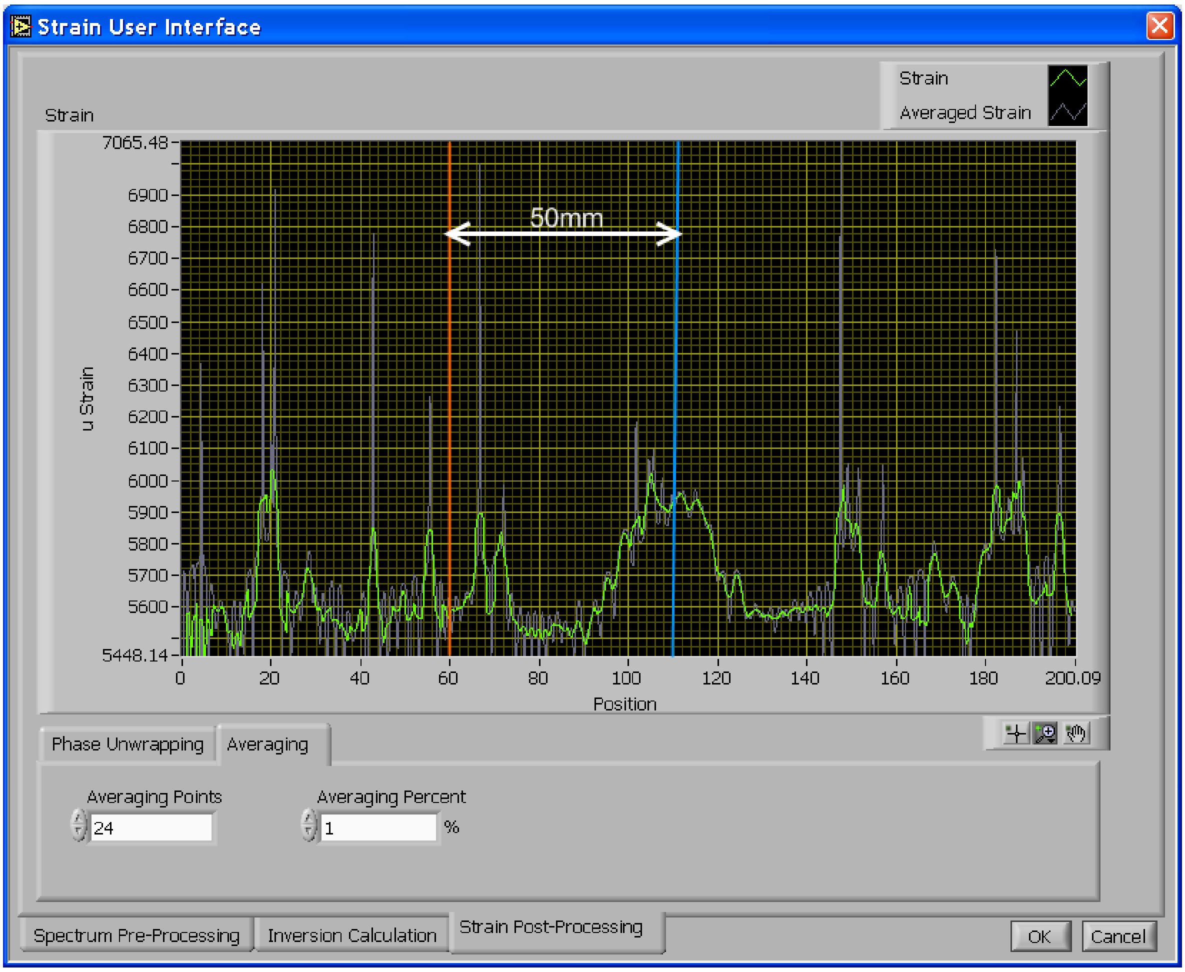


Figure 7.2.1: An extended inversion of a reflectance spectra to attempt to find the grating start position

8. Acknowlegments

Chris Brooks was an Industrial Experience Student from Swinburne University working with the Smart Structures and Advanced Diagnostics group on fibre optic sensing during a 12 month placement in 2005. The work described in this report was carried out as part of his industrial experience placement at DSTO.

Appendices

The automated spectra recording and analysis software described in this manual was written using the LabVIEW graphical programming package version 6.1. The following appendices show the interfaces and code for the software.

Appendix A: Hardware Interface

The Strain Measuring System software interfaces with an HP 8164A and optionally a JDS SB series optical switch. Both these devices used the GPIB interface. Communication with the devices is handled by the **GPIB Send** and **GPIB Query** VIs, for sending and recieving data respectively from the the devices.

A.1. Configuration

The software has been designed to be fully configurable in regards to the hardware interfaces. The configuration process allows the GPIB Bus and Address to be set for each devices and the Module and Channel numbers to be set for modules in the HP 8164A mainframe.

These values are saved to a settings file, allowing them to be reloaded each time the application starts. The first time the application starts, or if the settings file is deleted, it will load a list of default values, shown below in Table A.1.1.

Table A.1.1: Default Configuration Value

Settings	Default Value
8164A Bus	0
8164A Address	20
SB Switch Bus	0
SB Switch Address	10
TLD Slot	0
Grating Sensor Slot	1
Grating Sensor Channel	0
Laser Sensor Slot	1
Laser Sensor Channel	1
Interference Sensor Slot	2
Interference Sensor Channel	0
Reflection Switch Path	
Interference Switch Path	
Using SB Switch	False
Using Interference Power Sensor	False

The software is capable of automatically detecting connected devices. This process is performed when the user requests it. The search process queries all addresses on the specified GPIB bus and is discussed in more detail in section 1.4.

A.2. Initialisation

The Hardware Configuration steps are shown as a flow chart in Figure A.2.1. This section describes the steps performed.

1. When the application is started it will attempt to communicate with the hardware.
2. The application saves and loads configuration settings from a file named 'Settings.ini' in the save directory as the application.
3. If a settings file exists then the last used settings will be loaded from here.
4. If no saved settings can be found then a set of default values will be used (See Table A.1.1.).
5. Once the settings are loaded they are verified by attempting to communicate with the devices (See section A.3).
6. If all verifications are successful then the application will start. Otherwise the settings window will be displayed to allow the errors to be corrected.
7. Any value which caused an error is highlighted in red to draw attention to it.
8. As well as during start-up, the settings window is displayed by using the button in the main application window.
9. Display the settings window if it is not already shown.
10. The user can interact with the settings dialog, entering the necessary information.
11. If Offline mode is selected then there is no further interaction with the hardware. The application will start although it will be unable to perform scans.
12. Once the user has finished entering the configuration information it is verified as in step 5.
13. Hardware errors include the inability to communicate with a device or a device returning an unexpected ID.
14. Even if a hardware error is detected the application can still continue. The user is given the option to alter the value or continue using the entered value.
15. If the user wishes to change the values then the errors are marked and the settings window is displayed again.
16. Logical errors are caused by conflicting GPIB addresses or module/channel numbers. The two main causes are setting the mainframe and switch GPIB addresses to the same value and setting the same optical detector to measure different quantities. The application will not run if these errors are present.
17. Once the settings have been verified they are saved so they can be re-used next time the application starts.
18. The main interface is loaded once the settings window is closed by pressing either the *Offline Mode* button or the *Done* button.

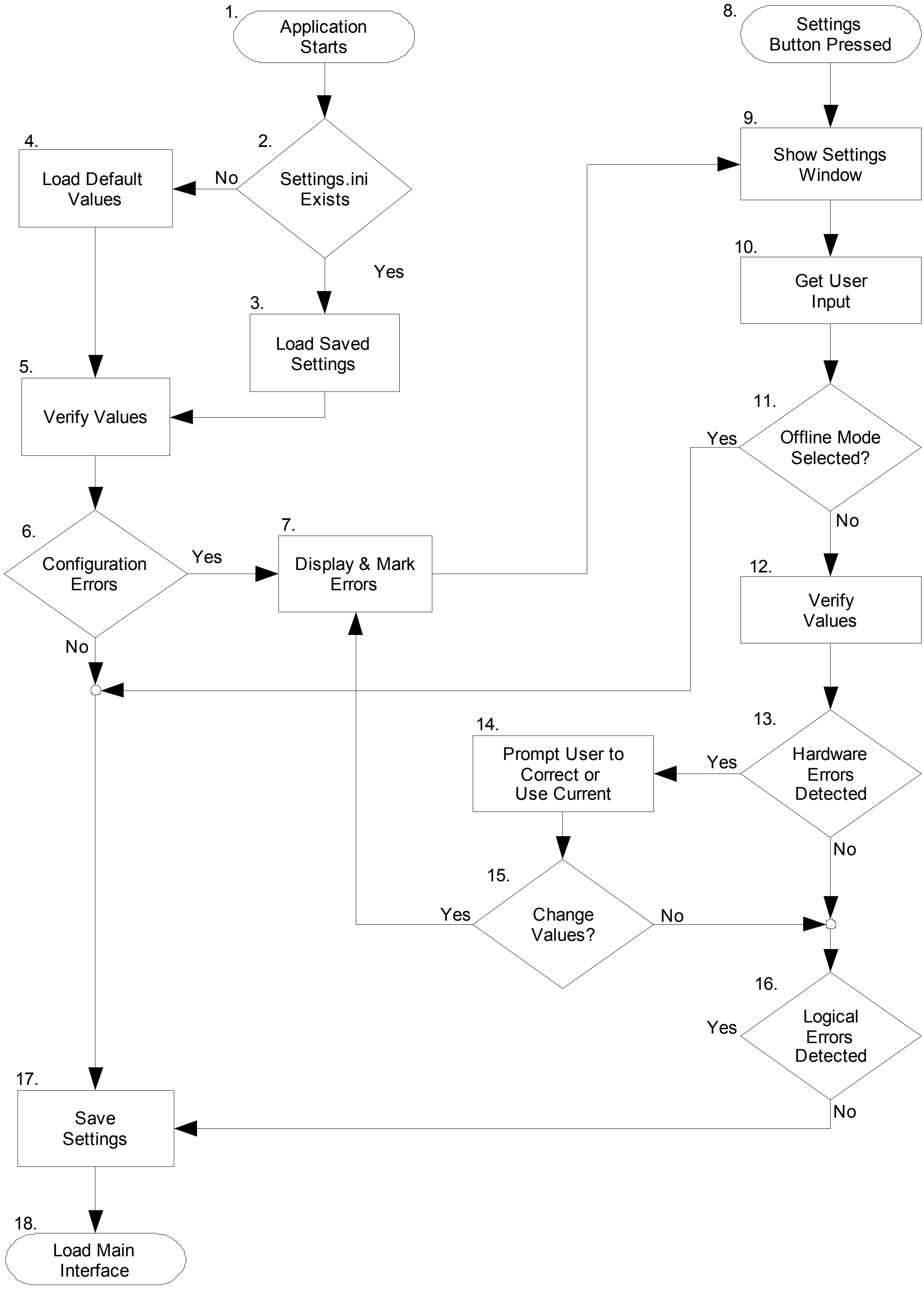


Figure A.2.1: Flow diagram for configuration initialization process

A.3. Device Verification

The process of verification of the hardware devices and configuration is shown in Figure A.3.1. The steps are described in more detail below.

1. Information about the HP8164A mainframe is retrieved by the **8164A Detector** VI.
2. The configuration settings are checked to determine if an SB series optical switch is to be used.
3. If an optical switch is set to be used then information about the device, including the number of available paths, is retrieved by the **SB Switch Detector** VI.
4. If an optical switch is used then the path numbers set for the reflection and interference paths are checked against the maximum number of paths available to the switch. If either of the numbers is too large then they are flagged.
5. The modules configured for the TLD and optical detector are compared against a list of module IDs compatible with the HP8164A mainframe. Any unknown or incompatible IDs are flagged. A module will be considered incompatible if, for example, an optical detector module is entered for the TLD slot number.
6. If any errors are flagged then a message box will be displayed informing the user.
7. The message box describes the error and presents the user with an option to either correct the value or to continue to use it. In most cases a value should be corrected however it is possible that an otherwise properly functioning device may return an incorrect identifier. Another possibility is that a new module may be used which has not been programmed into the system as being compatible.
8. The user may choose to either modify the values or continue using them.
9. No single module and channel combination can be used for more than one setting. For example the *Grating Power Sensor* and *Reflection Power Sensor* cannot both be set to the same value.
10. Any conflicts in module/channel assignments must be corrected.
11. Shows a message box informing the user of the conflicts.

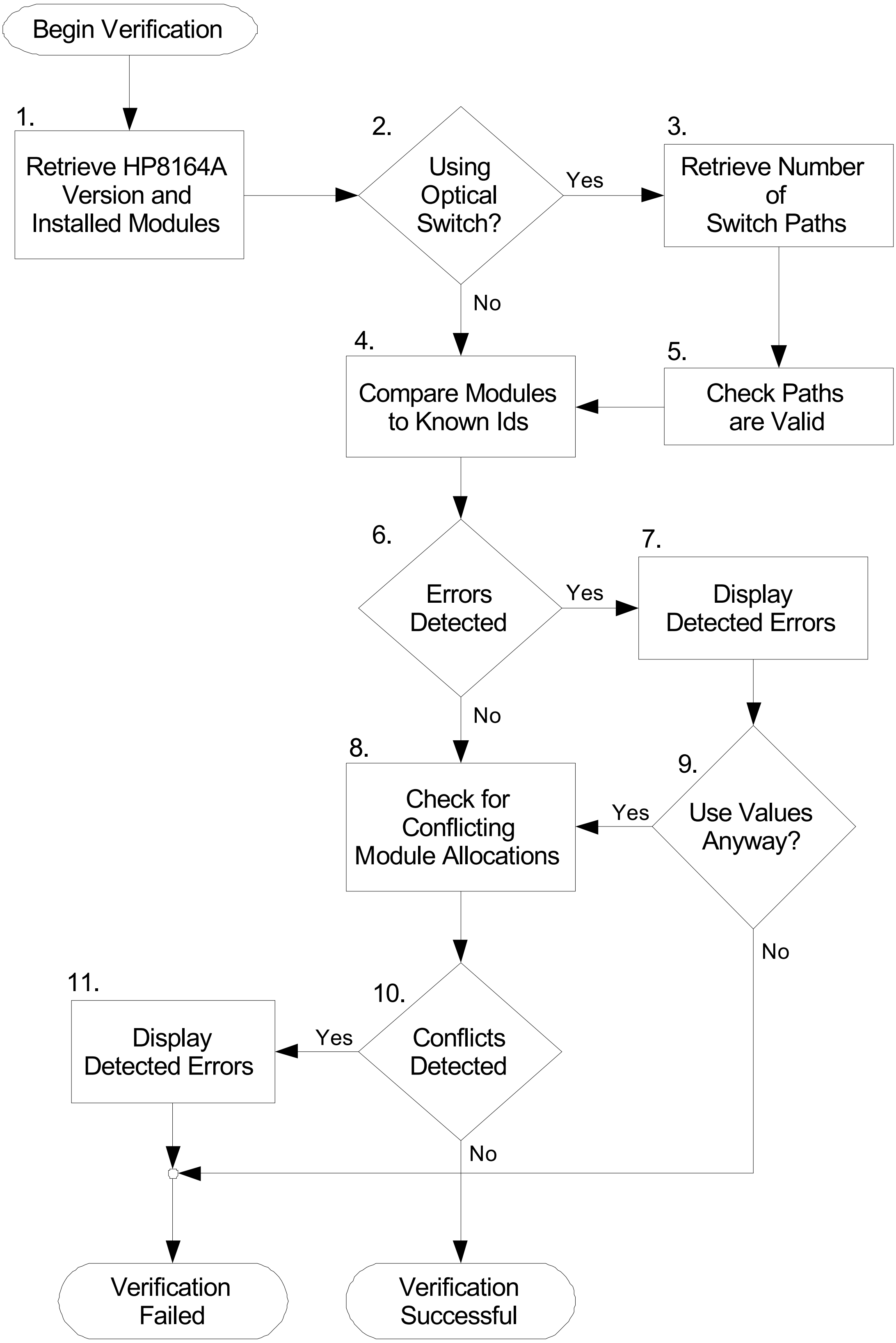


Figure A.3.1: Hardware Verification Process

A.4. Scanning

The steps taken when performing a spectrum scan are shown in Figure A.4.1. This section describes the steps in detail.

1. The scanning process is started by pressing the *Begin Sweep* button in the *Sweep Setup* tab.
2. If the *Scan Type* selection is set to 'Reflectivity' or 'Complete'(ie reflectivity and interference) then a reflectivity scan will be performed.
3. Checks if an optical switch is being used.
4. If an optical switch is being used then the reflection path is selected.
5. Perform a wavelength sweep. This process is detailed in Figure.
6. Stores the data returned from the wavelength sweep in the Reflectivity Wavelength, Reflectivity and Reflectivity Step data stores.
7. If the *Scan Type* selection is set to 'Interference' or 'Complete' then a interference scan will be performed.
8. Checks if an optical switch is being used.
9. If an optical switch is being used then the interference path is selected.
10. See Item 5.
11. Stores the data returned from the wavelength sweep in the Interference Wavelength, Interference Pattern and Interference Step data stores.
12. Attempt to detect the interference peaks using the 'Normalised Peak Detection' method. This step is detailed in section 3.2.2.
13. Once the required scans are completed the data is displayed on the appropriate graphs.

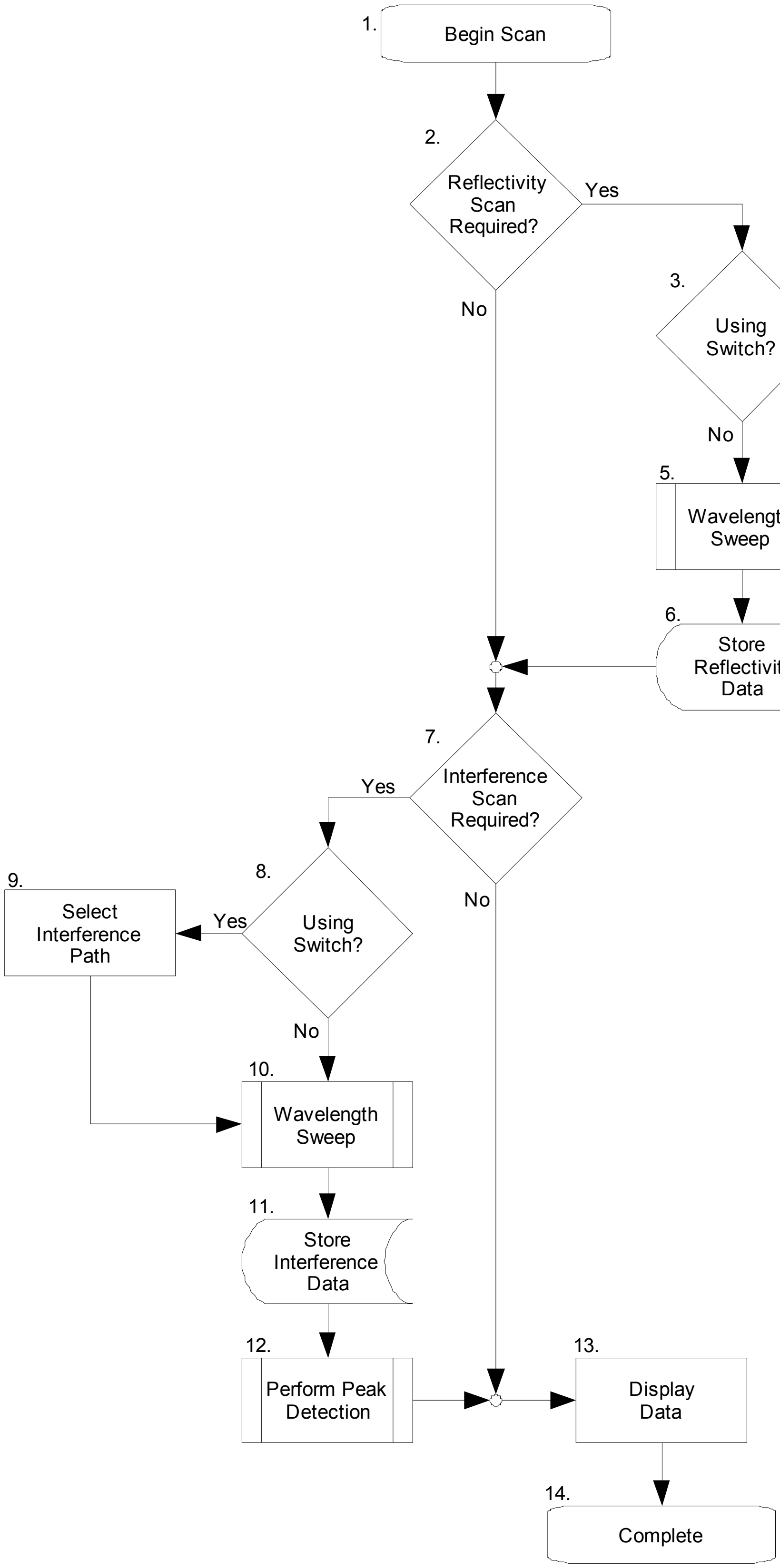


Figure A.4.1: Spectrum scan process

A.5. Wavelength Sweep

The steps taken when performing a wavelength sweep are shown in Figure A.5.1. This section describes the steps in detail.

1. Before setting the device parameters the wavelength range is first validated and confined between the values returned by the `source\textit{n}:wav? min` and `source\textit{n}:wav? max` GPIB queries.
2. The number of required data points, *n*, is calculated using the following formulae, where v_{sweep} is the sweep speed, λ_{range} is the scan wavelength range, λ_{step} is the scan step and f_{trig} is the scan triggering frequency.

$$f_{trig} = \frac{v_{sweep}}{\lambda_{step}}$$

$$\lambda_{step} = \frac{\lambda_{range}}{\min\left(f_{trig}, 10\mathrm{kHz}\right)}$$

$$n = \frac{\lambda_{range}}{\lambda_{step}}$$

3. Check if there are more points than the device is able to handle.
4. If $n > 16001$ then the new λ_{step} is calculated by repeatedly doubling λ_{step} until n is no more than 16001.
5. Before the sweep can begin certain parameters must be set for both the tuneable laser (TL) and the power meters. The settings are detailed in the table below. The bold values are values which are replaced by user set values and configuration settings.
   The averaging time, power range and wavelength are set and are not changeable in the application.
   The power range is selected to correspond to the maximum power available from the TLD. The maximum allowable power output from the HP 8164A mainframe is 15 mW, although not all modules will be able to reach this level.
   The sensor wavelength is set to the standard telecommunication wavelength, which most common fibre systems operate at. However altering this value appears to have little effect on the sensitivity over a large range of wavelengths.

Device	Parameter	Command
TLD	Wavelength Range Start	`source`**`n`**`:wav:sweep:start `**`Start`**`nm`
TLD	Wavelength Range End	`source`**`n`**`:wav:sweep:stop `**`End`**`nm`
TLD	Wavelength Sweep Speed	`source`**`n`**`:wav:sweep:speed `**`Speed`**`nm/s`
TLD	Wavelength Sweep Step	`source`**`n`**`:wav:sweep:step `**`Step`**`nm`
TLD	Wavelength Sweep Mode	`source`**`n`**`:wav:sweep:mode continuous`
TLD	Wavelength Sweep Cycle	`Source`**`n`**`:wav:sweep:cycles 1`
Detector	Averaging Time	`Sense`**`n`**`:power:atime 100us`
Detector	Manual Power Range	`Sense`**`n`**`:power:range:auto 0`
Detector	Channel 1 Wavelength	`Sense`**`n`**`:channel1:power:wav 1550nm`
Detector	Channel 2 Wavelength	`Sense`**`n`**`:channel2:power:wav 1550nm`
Detector	Channel 1 Power Range	`Sense`**`n`**`:channel1:power:range 10dBm`
Detector	Channel 2 Power Range	`sense`**`n`**`:channel2:power:range 10dBm`

6. Lambda logging causes a wavelength measurement to be made simultaneously with any measurement from the power meters.

In order to alter any of the logging parameters, such as the number of points and the averaging time, the logging function must first be stopped.

Device	Parameter	Command
Mainframe	Output Trigger Type	`trig:output stf`
TLD	AM Modulation	`source`**`n`**`:am:state off`
TLD	Lambda Logging	`source`**`n`**`:wav:sweep:llogging on`
Detector	Input Trigger Action	`trig`**`n`**`:input sme`
Detector	Stop Lambda Logging	`sense`**`n`**`:function:state logg,stop`
Detector	Lambda Logging Parameters	`sense`**`n`**`:function:par:logging` **`DataPoints`**`,` → **`AveragingTime`**
Detector	Start Lambda Logging	`sense`**`n`**`:function:state logg,start`

7. The GPIB query `lock?` is used to determine the locked state of the laser.
8. The laser can be unlocked using the GPIB command `lock 0,` **`password`** if the user wishes.
9. If the unlock dialog is cancelled then the scan cannot continue as the laser cannot be turned on.
10. The laser is turned on using the `source`**`n`**`:pow:state 1` command.
11. The scan processes is started using the `source`**`n`**`:wav:sweep:state start` command.
12. The software sends a `source`**`n`**`:wav:swe:state?` query every 250 ms. If the query returns 0 then the scan is complete.
13. The laser is turned off using the `source`**`n`**`:pow:state 0` command.
14. Data is retrieved from the device using the `source`**`n`**`:read:data? llog` command to retrieve the wavelength data and the `sense`**`n`**`:chanm:func:res?` command to retrieve the power data. from each power detector.

    Data from the HP8164A is ordered in reverse to the least significant byte first used in LabVIEW. The byte ordering must first be reversed before the data can be used.

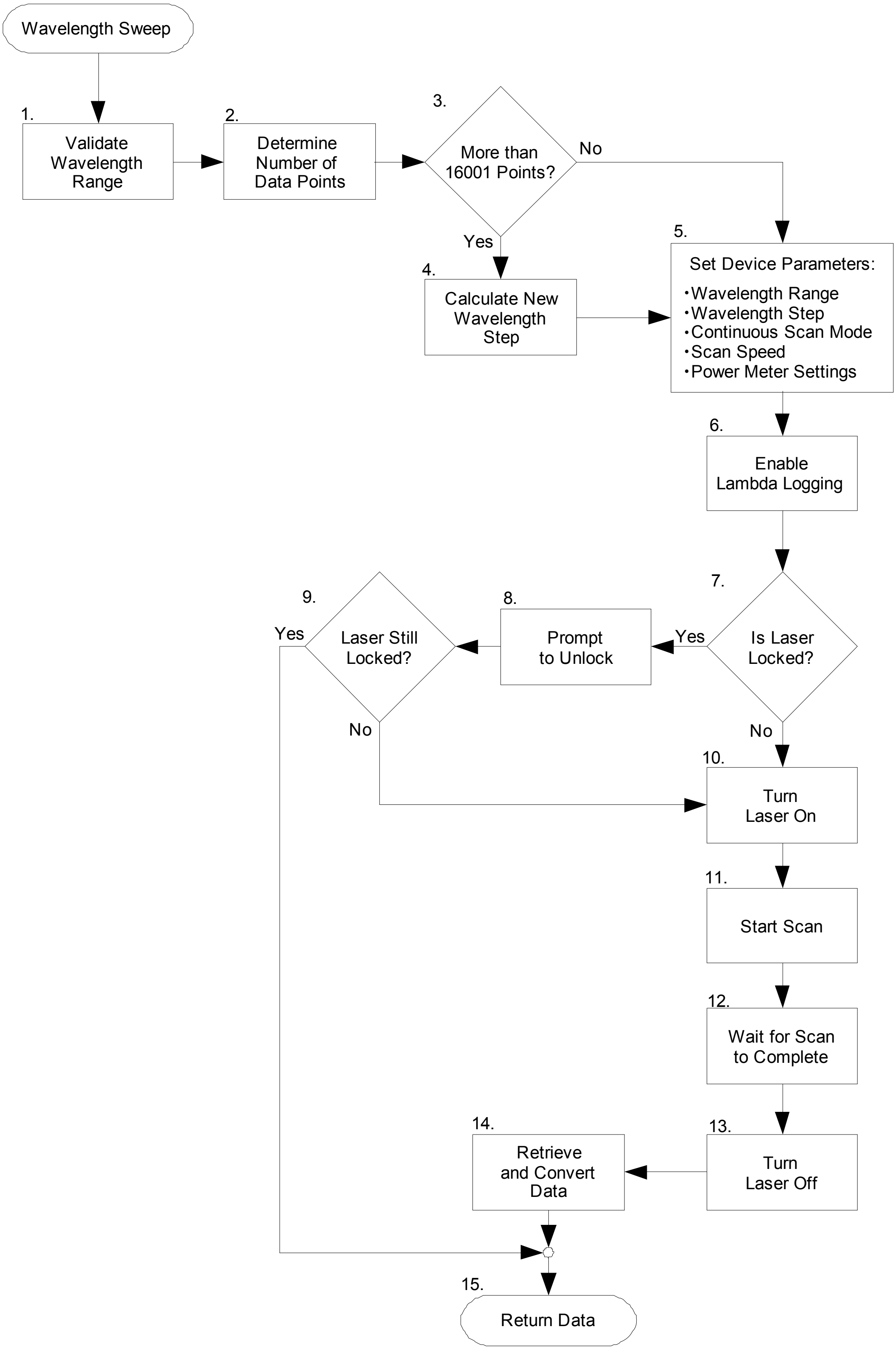


Figure A.5.1: Wavelength sweep process

A.6. Simple Peak Detection Algorithm

The Simple Peak Detection Algorithm finds all peaks in the interference spectrum and then removes those it determines to be 'ripples', depending on a user supplied threshold value. Figure A.6.1 shows the steps taken during the detection process, these steps are described in further detail below.

1. The average $\delta\lambda$ is calculated by dividing the wavelength range by the number of points minus one. The minimum $\delta\lambda$ from the reflectivity spectrum and interference spectrum is used.
2. The points in both the interference and reflectivity spectra are equally spaced using the minimum $\delta\lambda$ calculated in step 1.
3. The maxima and minima are located by comparing each intensity value to the previous one. Depending on whether a minimum or maximum was detected last the following conditions are checked:
   - If the last detected feature was a minimum, and the current point is less than the last point, then the last point was a maximum.
   - If the last detected feature was a maximum, and the current point is greater than the last point, then the last point was a minimum.
4. The number of minima and maxima found are stored for later use.
5. The Threshold is applied by comparing the difference between a point and the two minima or maxima to either side. If a point is a maximum then it will be surrounded by two minima, likewise a minimum will be surrounded by two maxima.
   If either of the surrounding minima or maxima are separated in the intensity scale by less than the value of threshold then the central point will be removed.
6. If the number of minima or maxima has changed after removing maxima or minima under the threshold then the procedure is applied again as this is the easiest way to ensure all ripples are removed. A single pass may not remove all ripples, especially if there are multiple ripples in a short wavelength range.

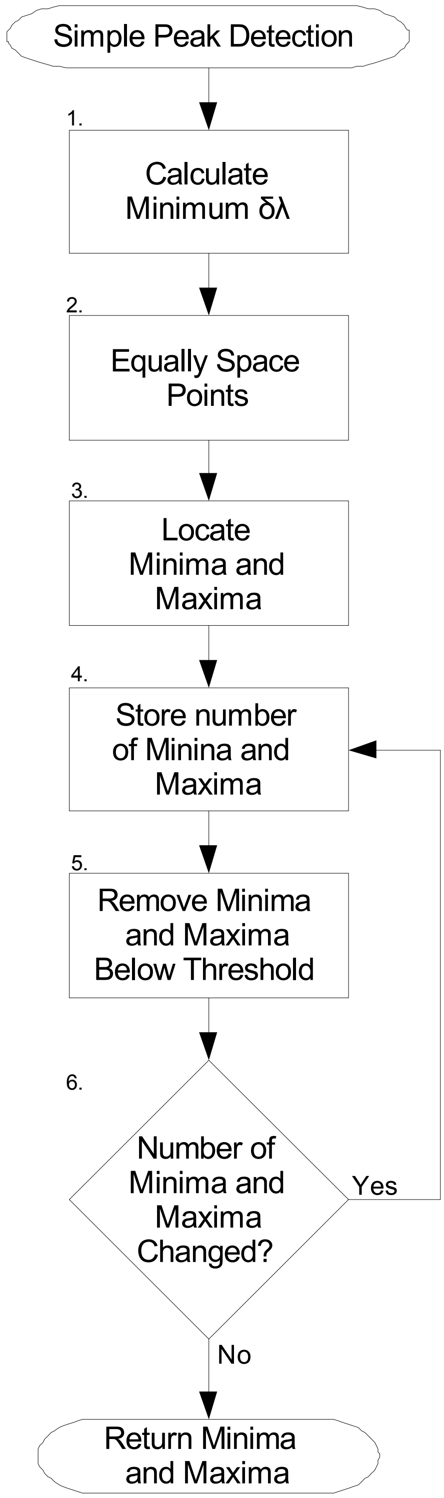


Figure A.6.1: Simple peak detection process

A.7. Normalized Peak Detection Algorithm

The Normalized Peak Detection Algorithm works by dividing the interference spectrum by an adjusted reflectance spectrum. This causes the peaks in the lower reflectivity regions of the spectrum to become larger and therefore less likely to be discarded as noise 'ripples'.

A threshold value is used to remove unwanted noise. During the detection process this value is used to create two thresholds, a positive and a negative. Both thresholds have an absolute value equal to the threshold supplied by the user.
Figure A.7.1 shows the steps taken during the detection process, these steps are described in further detail below.

1. The reflection and interference spectra are equally spaced at either the specified scan resolution or, when loaded from a saved file, the average wavelength spacing.
2. The reflection spectrum is overlaid against the interference spectrum. This is done using the following formula, where **R** is the reflection spectrum, **I** is the interference spectrum, *M* is the mode of the interference spectrum, taken using 500 levels and *S* is a scaling factor.
$$\mathbf{R}_{\text{Adjusted}} = \frac{(\mathbf{R} - \min \mathbf{R})(\max \mathbf{I} - M)}{(\max \mathbf{R} - \min \mathbf{R})S} + M$$
3. The interference spectrum is then adjusted using the adjusted reflection spectrum. This is done using the following formula, where *O* is a user adjusted offset.
$$\mathbf{I}_{\text{Adjusted}} = \frac{\mathbf{I} - (\mathbf{R}_{\text{Adjusted}} - O \max \mathbf{I})}{\mathbf{R}_{\text{Adjusted}}}$$
4. The initial settings for the peak detecting algorithm are to move through the wavelengths in a positive direction and to search for a maximum value.
5. The algorithm runs until it reaches the end of the interference spectrum array.
6. Checks if any values since the last detected maximum/minimum have passed the first threshold. If the algorithm is searching for a maximum then to pass the first threshold a value must be greater than the positive threshold value. If the algorithm is searching for a minimum then to pass the first threshold a value must be less than the negative threshold value.
7. If the first threshold has not yet been passed then the current point is compared against the appropriate threshold, as described in step 6.
8. If the current point passes the first threshold test then this is recorded.
9. If the first threshold has been passed then the current point is compared against the second threshold condition. If the algorithm is searching for a maximum then to pass the second threshold a value must be less than the negative threshold value. If the algorithm is searching for a minimum then to pass the second threshold a value must be greater than the positive threshold value.
10. If the second threshold condition is satisfied then the algorithm has detected a maximum or minimum. It will use the highest/lowest point recorded since last detection as the maximum or minimum point to be returned to the application. The search mode is then toggled between maximum and minimum.

11. The first threshold passed record is reset and the highest/lowest point records are also cleared.
12. Update the highest/lowest point record if necessary.
13. Moves to the next point in the spectrum. This either increases or decreases the wavelength depending on the current mode.
14. If the end of the spectrum is reached then the current direction mode is checked since the algorithm must run through the spectrum in both directions.
15. If the direction mode was set to forward then it is reversed and the algorithm repeated in the negative wavelength direction.

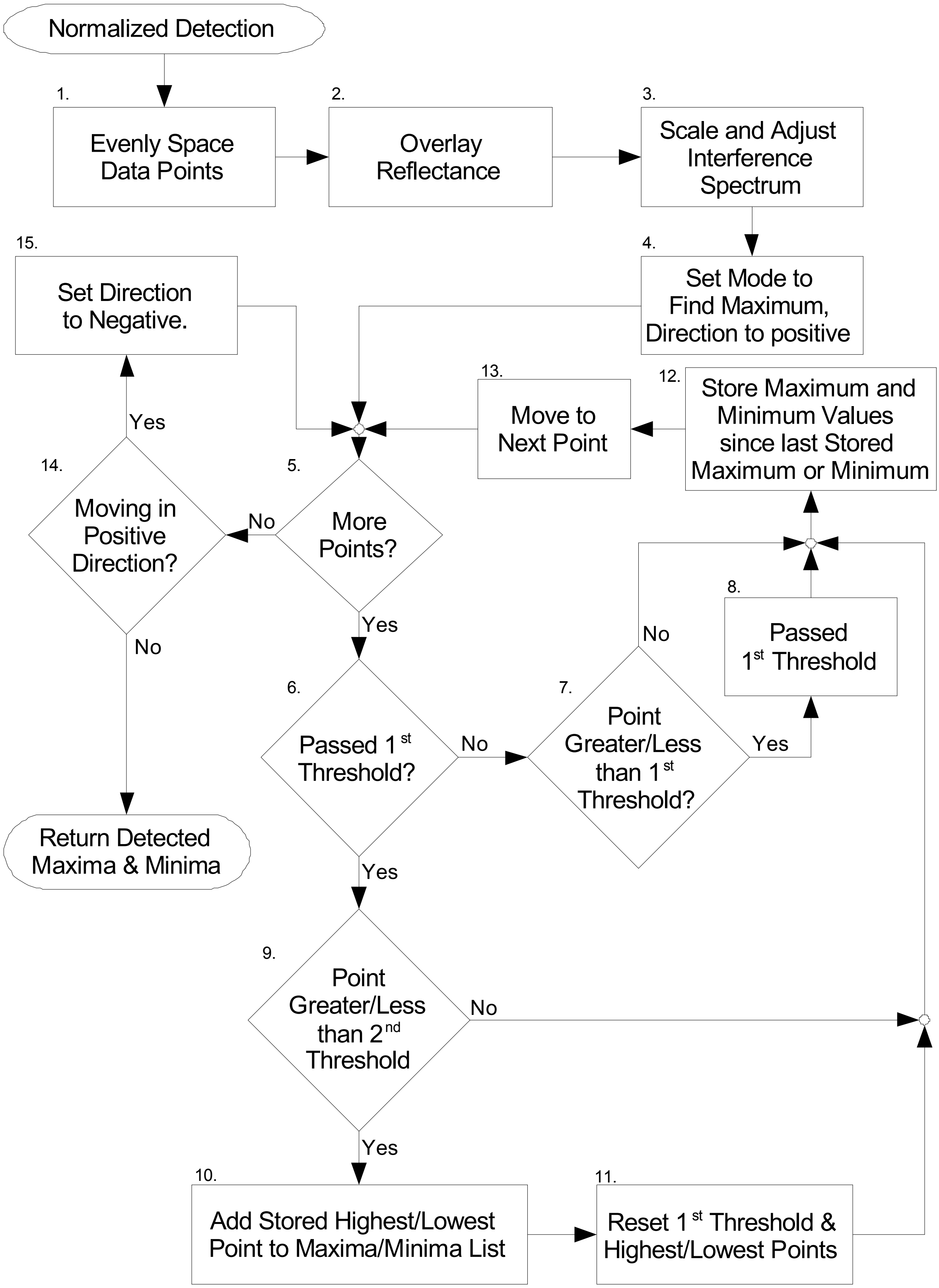


Figure A.7.1: Normalized peak detection process

A.8. Strain Calculation/Discreet Layer Peeling

The strain calculation process is a fairly linear series of steps and does not require a flow chart to aid in its explanation.

A.8.1 Spectrum Pre-processing

1. Evenly space the spectra data points using a user supplied number, which defaults to the average reflection spectrum point spacing.
2. Add or remove data points from both the reflection and phase spectra as required for the start and end wavelengths to be equal to the user specified values.
3. Convert the reflection and phase spectra into a complex reflection spectrum.

A.8.2 Grating Inversion using Discreet Layer Peeling

1. Convert from wavelength to frequency detuning domain using the following formula, where n is the refractive index of the fibre and λ_0 is the unstrained peak wavelength of the grating.
$$\delta(\lambda) = 2\pi n\left(\frac{1}{\lambda} - \frac{1}{\lambda_0}\right)$$
2. Calculate position step value, Δz using the following formula:
$$\Delta z = \frac{\pi}{\max\delta - \min\delta}$$
3. The partial complex reflection spectrum at each position, defined as $r(z,\delta)$ is initially set to equal the complex reflection spectrum of the grating at position $z = 0$.
4. The following equations are calculated for each z position between 0 and the specified grating length, L. ρ is the complex reflection coefficient and q is the coupling coefficient.
$$\rho(z) = \frac{1}{M}\sum_{m=1}^{M} r(z,\delta_m)$$
$$q(z) = -\frac{\tanh^{-1}(|\rho(z)|)\rho^*(z)}{\Delta_z|\rho(z)|}.$$
$$r(z+\Delta z,\delta) = \frac{r(z,\delta) - \rho(z)}{1-\rho^*(z)r(z,\delta)}\exp(-i2\delta\Delta z)$$
5. The strain profile is calculated from the derivative of the phase of q. However, the phase must first be unwrapped. This can be done by a threshold approach, in which any jump over a certain threshold is treated as wrapped and is unwrapped. The other approach is to force a positive or negative gradient. In this case any movement against this trend is treated as a wrapping.

6. Once the strain profile is calculated by taking the derivative of the q profile it is scaled using the formula below, where φ_q is the phase of q and v is Poisson's ratio and p_{11} and p_{12} are strain-optic coefficients of the fibre optic material.
$$a = 1 - \tfrac{1}{2} n_0^2 \left[p_{12} - v\left(p_{11} + p_{12}\right)\right]$$
$$\varepsilon(z) = -\frac{d\varphi_q(z)}{dz}\frac{\Lambda_0}{2\pi a}$$
7. Once the strain profile is calculated it may be averaged if desired by the user to attempt to smooth out noise and large spikes which may be present in the profile.

Appendix B: Data Storage

B.1. Internal Storage

This section describes the data storage variables used in the 'Strain Measurement Controller.vi' LabVIEW VI. These variables are used to store data between the different stages of processing.

Reflectivity Wavelength
Type: 1-D Double Array
Units: m
The wavelengths of each data point returned from a reflectivity scan.

Reflectivity
Type: 1-D Double Array
Units: None
The reflectivity of each data point returned from a reflectivity scan.

Reflectivity Step
Type: Double
Units: m
The specified step size of the reflectivity scan. This is not always exactly the same as the wavelength spacing of the data points.

Interference Wavelength
Type: 1-D Double Array
Units: m
The wavelengths of each data point returned from an interference scan.

Interference Pattern
Type: 1-D Double Array
Units: W
The power of each data point returned from an interference scan.

Interference Step
Type: Double
Units: m
The specified step size of the interference scan. This is not always exactly the same as the wavelength spacing of the data points.

Phase Wavelength
Type: 1-D Double Array
Units: m
The wavelength of each point in the phase profile.

Phase
Type: 1-D Double Array
Units: rad
The phase of each point in the phase profile.

Strain Position
Type: 1-D Double Array
Units: m
The position from the start of the grating of each point in the strain profile.

Strain Profile
Type: 1-D Double Array
Units: None
The strain at each point in the strain profile.

Adjusted Strain Position
Type: 1-D Double Array
Units: m
The position from the start of the grating of each point in the processed strain profile.

Adjusted Strain Profile
Type: 1-D Double Array
Units: None
The strain at each point in the processed strain profile.

Baseline Strain Position
Type: 1-D Double Array
Units: m
The position from the start of the grating of each point in the strain profile baseline.

Baseline Strain Profile
Type: 1-D Double Array
Units: None
The strain at each point in the processed strain profile baseline.

Manual Peaks
Type: 1-D 32-bit Integer Array
Units: N/A
Stores the index in the 'Interference Wavelength' and 'Interference Pattern' arrays of each manually selected peak.

Last Saved Data File
Type: File Path
Units: N/A
Stores the file name of the last file which was saved.

Last Loaded Data File
Type: File Path
Units: N/A
Stores the file name of the last file which was loaded.

B.2. Saved File Format

The application saves and loads data files using tab separated text files. This format is used for its compatibility between many different applications. Figure 7.2.1 shows the basic layout of the files. A list of compatible X and Y quantities are listed in Table 7.2.2, the names are not case sensitive. If desired all available Y quantities can be listed in the file, although only certain ones will be loaded back into the Strain Measuring System.

X Quantity	Tab	Y Quantity	Tab	X Quantity	Tab	Y Quantity	New Line
Data	Tab	Data	Tab	Data	Tab	Data	New Line
Data	Tab	Data	Tab	Data	Tab	Data	New Line
Data	Tab	Data	Tab	Data	Tab	Data	New Line

Figure 7.2.1: Sample of Tab Separated Format

Table 7.2.2: Saved quantities

Y Quantity	Associated X Quantity	Can be Loaded?
reflectivity	wavelength	Yes
interference	wavelength	Yes
phase	wavelength	Yes
strain	position	No
averaged strain	position	No

Appendix C: VI Reference

This section lists all the subVIs used by the strain measuring software. **Bold** Inputs are required and Grey inputs are optional, as they are displayed in LabVIEW.
The spectrum and profile data is, in most cases, passed as two separate equal length arrays, one containing the x-axis data (wavelength or position) and the other containing the y-axis data (such as reflectivity or strain).

C.1. Data Processing.llb

C.1.1 Double Byte-Swap.vi

The wavelength data returned from the HP 8164A has its bytes arranged in reverse to how LabView expects. This VI reverses the ordering of the bytes in a double value.

Inputs

Name	Type
Input Double	Double

Outputs

Name	Type
Output Double	Double

C.1.2 Remove Last Character.vi

Removes the last character from a string.

Inputs

Name	Type
Input	String

Outputs

Name	Type
Output	String

C.1.3 Single Byte-Swap.vi

The power data returned from the HP 8164A has its bytes arranged in reverse to how LabVIEW expects. This VI reverses the ordering of the bytes in a single value.

Inputs

Name	Type
Input Single	Single

Outputs

Name	Type
Output Single	Single

C.1.4 Trim String.vi

Removes all whitespace (as defined by the 'White Space?' VI) from the beginning and end of a string.

Inputs

Name	Type
String	String

Outputs

Name	Type
Trimmed String	String

C.2. Grating Inversion.llb

C.2.1 DLP Inversion

This VI performs a Discreet Layer Peeling inversion on the complex reflection spectrum of a grating to calculate the coupling coefficient and strain profile the grating is placed under.
The Wavelength and Reflectance arrays must both be the same length.

Inputs

Name	Type	Units	Default
Wavelength	Double Array		
Reflectance	Complex Double Array		
Grating Length	Double	m	
Lambda 0	Double	m	1550e-9
Grating Index	Double	m	5e-3
Grating Start	Double	m	0

Outputs

Name	Type	Units
Residual	Double	
Position	Double Array	m
Strain Profile	Double Array	
Coupling Coefficient	Complex Double Array	/m

C.2.2 Spectrum Pre-Processing

This VI performs the pre-processing of the reflectance spectrum before it is inverted. It equally spaces the data points, adds/removes data to satisfy the start and end wavelength specifications and then combines the reflectivity and phase spectra into a complex reflectance spectrum by interpolating the phase values.

The Used Phase Interpolation output returns the interpolation used as occasionally the Spline interpolation will fail if a wavelength array passed to it has a duplicated value.

The Input Reflectivity Wavelength and Input Reflectivity arrays should both have the same number of entries as should the Input Phase Wavelength and Input Phase arrays.

Inputs

Name	Type	Units	Default
Phase Interpolation	Enumeration		Spline
Input Reflectivity Wavelength	Double Array	m	
Input Reflectivity	Double Array		
Input Data Point Spacing	Double	m	
Input Phase Wavelength	Double Array	m	
Input Phase	Double Array	rad	
Start Wavelength	Double	m	
End Wavelength	Double	m	
Added Data Values	Enumeration		Last Value
Error In	Error Cluster		No Error

Outputs

Name	Type	Units	Default
Pre-Processed Wavelength	Double Array	m	
Pre-Processed Reflectance	Complex Double Array		
Used Interpolation	Enumeration		
Error Out	Error Cluster		

C.3. HP 8164A.llb

C.3.1 8164A Detector

This VI attempts to detect an HP 8164A mainframe at the specified address. If the detection is successful it will return two clusters containing hardware information and the IDs of any installed modules.

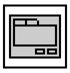
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Error In	Error Cluster	

Outputs

Name	Type
Instrument ID	**Cluster**
Manufacturer	String
Instrument Model	String
Serial Number	String
Firmware Version	String
Installed Modules	**Cluster**
Slot 0	String
Slot 1	String
Slot 2	String
Slot 3	String
Slot 4	String
Found HP8164A	Boolean
Error Out	Error Cluster

C.3.2 Configuration



This VI controls the hardware configuration settings. It allows the user to specify the addresses and module numbers and then attempts to verify them. A more detailed explanation of how this VI works can be found in appendix A.3. When loaded, this VI will load any previously saved settings from the application's INI file. All settings are output as one cluster.

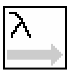
Inputs

Name	Type	Default
Show Dialog	Boolean	False

Outputs

Name	Type
Configuration Settings	Cluster
Config Errors	Boolean

C.3.3 Continuous Sweep Setup



This VI configures an optical source in the HP 8164A mainframe for a continuous wavelength sweep. If a Trigger Step of 0 is used then the VI will determine the smallest allowable value and use that.

Inputs

Name	Type	Units	Default
GPIB Bus	32-bit Integer		0
GPIB Address	16-bit Integer		
Sweep Speed	Enumeration		No Change
Trigger Step	Double	m	0
Module Number	16-bit Unsigned Integer		0
Sweep Cycles	16-bit Unsigned Integer		1
Error In	Error Cluster		No Error

Outputs

Name	Type	Units
Sweep Parameters	**Cluster**	
Module Number	16-bit Integer	
Sweep Speed	Enumeration	
Trigger Step	Double	m
Error Out		

C.3.4 GPIB Query

This VI sends a query message to a hardware device and then waits for and returns its response. The response will be terminated by the specified Terminating Character.

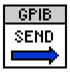
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Maximum Bytes Returned	32-bit Integer	255
Query String	String	
Error In	Error Cluster	No Error
Terminating Character	16-bit Integer	0

Outputs

Name	Type
Query Result	String
Bytes Returned	32-bit Integer
GPIB Status	Boolean Array
Error Out	Error Cluster

C.3.5 GPIB Send



This VI sends a command to a hardware device.

Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Command String	String	
Error In	Error Cluster	No Error

Outputs

Name	Type	Default
Bytes Sent	32-bit Integer	
GPIB Status	Boolean Array	
Error Out	Error Cluster	No Error

C.3.6 Lambda Logging Setup

This VI configures the 8164A mainframe and a specified optical source to either enable or disable lambda logging, which causes every power measurement to also record the wavelength of the laser at that exact moment.

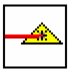
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Enable Lambda Logging	Boolean	True
Module Number	16-bit Unsigned Integer	0
Error In	Error Cluster	No Error

Outputs

Name	Type
Error Out	Error Cluster

C.3.7 Laser State



This VI can turn an optical source on or off. If the laser lock is activated it will prompt the user for the password to unlock the laser. It will return the current state of the laser, which may not have changed if the laser was not unlocked.
A requested state of 'Query' can be passed to retrieve the current state of the laser.

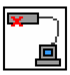
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
State	Enumeration	Query
Module Number	16-bit Unsigned Integer	0
Error In	Error Cluster	No Error

Outputs

Name	Type
Laser On	Boolean
Error Out	Error Cluster

C.3.8 Lockout Local Control



This VI locks local control from a hardware device. This is done during the wavelength sweep process to prevent it from being interrupted.

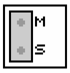
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Error In	Error Cluster	No Error

Outputs

Name	Type
Error Out	Error Cluster

C.3.9 Power Meter Setup



This VI configures the settings of a dual power sensor module.
The automatic power range can only be set on a per-module basis. If the master range is set to automatic then the slave range will also be set to automatic. If the slave range is set to automatic and the master range is given a value then the slave range will be set to the master's range value.

Inputs

Name	Type	Units	Default
Averaging Time	Double	s	1e-3
GPIB Bus	32-bit Integer		0
GPIB Address	16-bit Integer		
Master Power Range	Enumeration		Auto
Slave Power Range	Enumeration		Auto
Module Number	16-bit Unsigned Integer		1
Error In	Error Cluster		No Error
Master Wavelength	Double	m	1550e-9
Slave Wavelength	Double	m	1550e-9

Outputs

Name	Type
Error Out	Error Cluster

C.3.10 Power Reader

This VI returns the recorded power data from a detector. This data is then converted to a format compatible with LabVIEW.

Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Module Number	16-bit Unsigned Integer	1
Channel	Enumeration	Master
Error In	Error Cluster	No Error

Outputs

Name	Type	Units
Power	Double Array	W
Error Out	Error Cluster	

C.3.11 Power Sensor Reading Setup

This VI sets the parameters of an optical detector module used during a wavelength sweep. It will automatically adjust the trigger step to ensure the number of returned points does not exceed the mainframe's limit of 16001.
The Sweep Parameters input should be connected to the Sweep Parameters output of the 'Continuous Sweep Setup' VI.

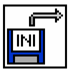
Inputs

Name	Type	Units	Default
GPIB Bus	32-bit Integer		0
GPIB Address	16-bit Integer		
Averaging Time	Double	s	100e-6
Module Number	16-bit Unsigned Integer		1
Error In	Error Cluster		No Error
Sweep Parameters	Cluster		

Outputs

Name	Type	Units
Data Points	32-bit Integer	
Sample Rate	Double	Hz
Sweep Parameters Out	Cluster	
Error Out	Error Cluster	

C.3.12 Read Configurations



This VI reads saved configuration data from the 'Settings.ini' file in the application directory. If this file cannot be found then a set of default values will be used. The configuration settings are returned as a single cluster due to their large number.

Inputs

Name	Type	Default
Error In	Error Cluster	No Error

Outputs

Name	Type
Configuration Settings	Cluster
Error Out	Error Cluster

C.3.13 Retrieve Module Data

This VI retrieves the power data for a specified channel and module from an array of compounded power data. The array is an array of clusters containing the Master and Slave data from each module.

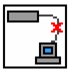
Inputs

Name	Type	Default
Slot Number	16-bit Unsigned Integer	
Channel Number	32-bit Integer	0
Power Readings	Cluster Array	

Outputs

Name	Type	Units
Power Data	Double Array	W

C.3.14 Return to Local



Returns control to the specified hardware device. This undoes the operation performed by the 'Lockout Local Control' VI.

Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Error In	Error Cluster	No Error

Outputs

Name	Type
Error Out	Error Cluster

C.3.15 Run Sweep

This VI performs a wavelength sweep and retrieves the wavelength data and power data from a list of specified modules and channels. This data is made uni-directional and then compounded into an array of clusters. Individual readings can be retrieved from this compound array using the 'Retrieve Module Data' VI.
If the laser cannot be unlocked then the scan can not be completed.

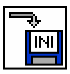
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Module Numbers	**Cluster**	
Laser Module	16-bit Unsigned Integer	
Power Sensor Modules	**Cluster Array**	
Sensor Slot	16-bit Unsigned Integer	
Sensor Channel	32-bit Integer	
Error In	Error Cluster	No Error

Outputs

Name	Type
Wavelengths	Double Array
Power Readings	Cluster Array
Scan Complete	Boolean
Error Out	Error Cluster

C.3.16 Save Configurations



This VI saves the hardware configuration to the 'Settings.ini' file in the application directory so they can be reused next time the application is loaded.

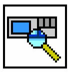
Inputs

Name	Type	Default
Configuration Settings	Cluster	
Error In	Error Cluster	No Error

Outputs

Name	Type
Error Out	Error Cluster

C.3.17 Search for Devices



This VI attempts to automatically find the GPIB address of an HP 8164A mainframe, and optionally a JDS SB Series switch, on a given GPIB Bus.
It performs the search by querying each address from 0 to 30 and checking the ID response against that expected from either device.
If a device cannot be found then its address will be reported as -1.

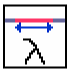
Inputs

Name	Type	Default
GPIB Bus to Search	32-bit Integer	0
Search for Switch	Boolean	False

Outputs

Name	Type
Bus Searched	32-bit Integer
8164A Address	16-bit Integer
Switch Address	16-bit Integer

C.3.18 Sweep Range



Ensures the start and end wavelengths specified for the sweep parameters are within the wavelength range of the hardware. The 'Validate Wavelength' VI is used to perform the checking.

Inputs

Name	Type	Units	Default
GPIB Bus	32-bit Integer		0
GPIB Address	16-bit Integer		
Start Wavelength	Double	m	
End Wavelength	Double	m	
Module Number	16-bit Unsigned Integer		1
Error In	Error Cluster		No Error

Outputs

Name	Type	Units
Start Wavelength	Double	m
End Wavelength	Double	m
Wavelength Changed	Boolean	
Error Out	Error Cluster	

C.3.19 Unlock Laser Dialog

Presents the user with a dialog box prompting for the unlock password for the laser. The dialog will stay until a valid password is entered or it is cancelled.

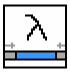
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Error In	Error Cluster	No Error

Outputs

Name	Type
Laser Unlocked	Boolean
Error Out	Error Cluster

C.3.20 Validate Wavelength



This VI forces the specified wavelength value to be within the allowed values of the hardware device. The allowable range of wavelengths is obtained from the 'Wavelength Range' VI.

Inputs

Name	Type	Units	Default
GPIB Bus	32-bit Integer		0
GPIB Address	16-bit Integer		
Wavelength	Double	m	
Module Number	16-bit Unsigned Integer		0

Outputs

Name	Type	Units
Valid Value	Double	m
Value Changed	Boolean	

C.3.21 Wavelength Range

This VI queries a tuneable optical source module for the minimum and maximum wavelengths it is able to generate.

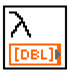
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Module Number	16-bit Unsigned Integer	0

Outputs

Name	Type	Units
Minimum Wavelength	Double	m
Maximum Wavelength	Double	m

C.3.22 Wavelength Reader



This VI returns the stored wavelength data recorded during a wavelength sweep with Lambda Logging enabled. The data is converted into a format compatible with LabVIEW.

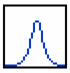
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Module Number	16-bit Unsigned Integer	0
Error In	Error Cluster	No Error

Outputs

Name	Type	Units
Wavelengths	Double Array	m
Error Out	Error Cluster	

C.3.23 Wavelength Sweep



This VI performs the necessary setup steps and then runs a wavelength sweep recording wavelength data and power data from a list of specified power meters.

The sweep will not be able to complete if the laser is not unlocked.

The Power Readings cluster is the same as from the 'Run Sweep' VI

Inputs

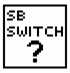
Name	Type	Units	Default
GPIB Bus	32-bit Integer		0
GPIB Address	16-bit Integer		
Sweep Speed	Enumeration		5nm/s
Sweep Wavelength	**Cluster**		
Start Wavelength	Double	m	
End Wavelength	Double	m	
Error In	Error Cluster		No Error
Module Numbers	**Cluster Array**		
Laser Module Slot	16-bit Unsigned Integer		
Power Sensor Modules	**Cluster Array**		
Slot	16-bit Unsigned Integer		
Channel	32-bit Integer		

Outputs

Name	Type	Units
Wavelength	Double Array	m
Power Readings	Cluster Array	
Scan Complete	Boolean	
Error Out	Error Cluster	
Scan Parameters	**Cluster**	
Sweep Step	Double	m
Sample Rate	Double	Hz
Scan Range	Double	m
Scan Time	Double	s
Data Points	32-bit Integer	

C.4. JDS SB Switch.llb

C.4.1 SB Switch Detector



This VI attempts detects a JDS SB Series optical switch and returns information on the device if successful.

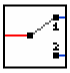
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Error In	Error Cluster	No Error

Outputs

Name	Type
Instrument ID	**Cluster**
Manufacturer	String
Instrument Model	String
Instrument Model (cont.)	String
Firmware Version	String
Found SB Switch	Boolean
Error Out	Error Cluster

C.4.2 Select Path



This VI attempts to select the specified path on an optical switch.

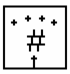
Inputs

Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Path	32-bit Integer	
Error In	Error Cluster	No Error

Outputs

Name	Type
Path Selected	Boolean
Error Out	Error Cluster

C.4.3 Switch Paths



This VI returns the number of available optical paths on an optical switch.

Inputs

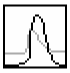
Name	Type	Default
GPIB Bus	32-bit Integer	0
GPIB Address	16-bit Integer	
Error In	Error Cluster	No Error

Outputs

Name	Type
Paths	32-bit Integer
Error Out	Error Cluster

C.5. Signal Processing.llb

C.5.1 Adjust Reflectance



This VI performs the following adjustments to a reflectance spectrum in the following order: Baseline Adjustment, Scaling, Reflectivity Coercion (between 0 and 1) and Wavelength range Clipping. The clipping is performed by the 'Clip to Range' VI.

The Input Wavelength and Input Reflectance arrays should be the same length.

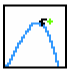
Inputs

Name	Type	Units	Default
Input Wavelength	Double Array	m	
Input Reflectance	Double Array		
Baseline Adjustment	Double		0
Scale Adjustment	Double		1
Wavelength Range	**Cluster**		
Minimum Wavelength	Double	m	0
Maximum Wavelength	Double	m	0

Outputs

Name	Type	Units
Output Wavelength	Double Array	m
Output Reflectance	Double Array	

C.5.2 Align Wavelengths



This VI aligns a set of maxima and minima wavelengths with those of the interference spectrum. The equalizing of the wavelength spacing performed during the detection process may result in peaks at wavelengths which don't have a corresponding data point in the interference spectrum.

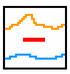
Inputs

Name	Type	Units
Maxima Wavelength	Double Array	m
Minima Wavelength	Double Array	m
Interference Wavelength	Double Array	m

Outputs

Name	Type	Units
Aligned Maxima	Double Array	m
Aligned Minima	Double Array	m

C.5.3 Array Subtraction



This VI subtracts an array of double values from another. It performs a spline interpolation on the second array to ensure both arrays have the same number of points. For this reason the VI will not work if the second array has any duplicated X values.

The X1 and Y1 arrays should have the same length, as should the X2 and Y2 arrays.

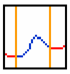
Inputs

Name	Type
X1	Double Array
Y1	Double Array
X2	Double Array
Y2	Double Array

Outputs

Name	Type
X1	Double Array
Y1 – Y2	Double Array

C.5.4 Clip to Range



This VI clips an array to ensure it does not extend beyond set x boundaries. If any of the boundaries specified are outside the x range of the array then they are ignored.
This VI is used mainly for clipping spectra, confining them to a specified wavelength range.
The X and Y arrays should be of the same length.

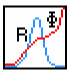
Inputs

Name	Type
X	Double Array
Y	Double Array
Lower X	Double
Lower Y	Double

Outputs

Name	Type
Clipped X	Double Array
Clipped Y	Double Array

C.5.5 Create Complex Reflection



This VI combines a reflectivity spectrum and a phase spectrum to create a complex reflectance spectrum. The complex spectrum, **r**, is formed from the equation below, where **R** is the reflectivity spectrum and **φ** is the phase spectrum.

$$\mathbf{r} = \mathbf{R}\exp(i\varphi)$$

Both the reflectivity and phase spectra must both have the same number of data points, this requires that the phase spectrum is interpolated and then extended at the lower and upper wavelength regions if required.
Occasionally a repeated wavelength value can appear in the data which prevents the spline interpolation from working. In this case the VI will fall back onto a linear interpolation.
The Intensity Wavelength and Intensity arrays should be the same length, as should the Phase Wavelength and Phase arrays.

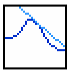
Inputs

Name	Type	Units	Default
Interpolation	Enumeration		Spline
Intensity Wavelength	Double Array	m	
Intensity	Double Array		
Phase Wavelength	Double Array	m	
Phase	Double Array	rad	

Outputs

Name	Type	Units
Wavelength	Double Array	m
Reflectance	Complex Double Array	
Used Interpolation	Enumeration	

C.5.6 Differentiate



This VI differentiates an array. The X and Y arrays should be the same length.

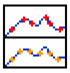
Inputs

Name	Type
X	Double Array
Y	Double Array

Outputs

Name	Type
dX/dY	Double Array

C.5.7 Equally Space Points



This VI equally spaces the points in an array. A spacing value and starting value can be specified or it can be automatically taken from the array if the passed value is not valid.

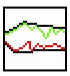
Inputs

Name	Type	Default
X	Double Array	
Y	Double Array	
X Start	Double	NaN
dX	Double	NaN

Outputs

Name	Type
Output X	Double Array
Output Y	Double Array

C.5.8 Find Filtered Peaks



This VI is used by the Normalized Peak Detection VI to find the peaks in a normalized interference spectrum. More details of how this VI operates can be found in Appendix A.6.

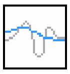
Inputs

Name	Type	Default
Power	Double Array	
Threshold	Double	0

Outputs

Name	Type
Maxima Indices	32-bit Integer Array
Minima Indices	32-bit Integer Array

C.5.9 Moving Average



This VI performs a moving average on an array. It can be set to automatically realign the data by shifting it forward. This prevents peaks from being pushed to the right as they are averaged out.

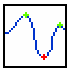
Inputs

Name	Type	Default
Input X	Double Array	
Input	Double Array	
Averages	32-bit Integer	2
Automatically Realign	Boolean	False

Outputs

Name	Type
Output X	Double Array
Output	Double Array

C.5.10 Noise Maxima & Minima



This VI finds the maxima and minima in an interference spectrum and then removes any which are below a threshold value to eliminate noise 'ripples' being detected as peaks. This process is described in more detail in Appendix A.5.

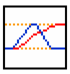
Inputs

Name	Type	Default
Y Values	Double Array	
Threshold	Double	0

Outputs

Name	Type
Maxima Indices	32-bit Integer Array
Minima Indices	32-bit Integer Array

C.5.11 Normalize and Overlay



This VI normalizes and overlays the second array on top of the first array. This is commonly used to display both reflectivity and phase on the same graph as they have greatly differing value ranges. Array 1 is not changed by this VI.

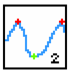
Inputs

Name	Type
Array 1	Double Array
Array 2	Double Array

Outputs

Name	Type
Output Array 1	Double Array
Output Array 2	Double Array

C.5.12 Peak Detector Simple



This VI performs a simple peak detection on an interference spectrum. More details of this process can be found in Appendix A.5.

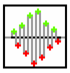
Inputs

Name	Type	Units	Default
Reflectance Wavelength	Double Array	m	
Reflectance	Double Array		
Interference Wavelength	Double Array	m	
Interference	Double Array	W	
Threshold	Double		0

Outputs

Name	Type	Units
Maxima Wavelengths	Double Array	m
Maxima	Double Array	W
Minima Wavelengths	Double Array	m
Minima	Double Array	W
Preview	X-Y Graph Cluster	

C.5.13 Peak Detector



This VI performs a normalized peak detection. More details on this process can be found in Appendix A.6.

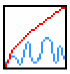
Input

Name	Type	Units	Default
Wavelength Step	Double		0.0001
Detection Offset	Double		0
Reflectivity Wavelength	Double Array	m	
Reflectivity	Double Array		
Interference Wavelength	Double Array	m	
Interference	Double Array	W	
Threshold	Double		0.5

Outputs

Name	Type	Units
Maxima Wavelength	Double Array	m
Maxima	Double Array	
Minima Wavelength	Double Array	m
Minima	Double Array	
Preview	X-Y Graph Cluster	

C.5.14 Peaks to Phase



This VI converts interference peaks into a phase spectrum. It requires that the maxima are specified but can also take a list of minima to improve the accuracy.

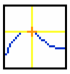
Inputs

Name	Type	Units	Default
Maxima Wavelength	Double Array	m	
Minima Wavelength	Double Array	m	Empty

Outputs

Name	Type	Units
Phase Wavelength	Double Array	m
Phase	Double Array	rad

C.5.15 Points to Cursors



This VI converts a list of points into an array of graph cursors. It is mainly used for displaying interference peaks on graphs.

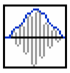
Inputs

Name	Type	Default
X Values	Double Array	
Y Values	Double Array	
Color	32-bit Unsigned Integer	0 (Black)
Point Style	8-bit Unsigned Integer	0
Hair Style	8-bit Unsigned Integer	0

Outputs

Name	Type
Cursor List	Graph Cursor Cluster Array

C.5.16 Reflectance Overlay



This VI overlays a reflection spectrum against an interference spectrum. The mode of the interference spectrum is taken as its baseline.

Inputs

Name	Type	Units	Default
Reflectance Wavelength	Double Array	m	
Reflectance	Double Array		
Interference Wavelength	Double Array	m	
Interference	Double Array	W	
Scale	Double		1

Outputs

Name	Type	Units
Reflectance Wavelength	Double Array	m
Adjusted Reflectance	Double Array	

C.5.17 Remove Complex NaNs

This VI removes any NaN values in a complex double array.

Inputs

Name	Type
Input Array	Complex Double Array

Outputs

Name	Type
Output Array	Complex Double Array

C.5.18 Remove NaNs

This VI removes any NaN values in a double array.

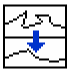
Inputs

Name	Type
Input Array	Double Array

Outputs

Name	Type
Output Array	Double Array

C.5.19 Unidirectional



This VI ensures that the X values of an array are always increasing by removing any values which are not.

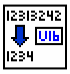
Inputs

Name	Type
X	Double Array
Y	Double Array

Outputs

Name	Type
Filtered X	Double Array
Filtered Y	Double Array

C.5.20 Unique Array Values (U16)



This VI removes any duplicated values from an array of unsigned 16-bit integers.

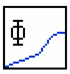
Inputs

Name	Type
Input	16-bit Unsigned Integer Array

Output

Name	Type
Output	16-bit Unsigned Integer Array

C.5.21 Unwrap Phase



This VI unwraps a phase profile so it is no longer constrained between $\pm 2\pi$. The unwrapping can be done to either force a positive or negative gradient, or can be done using a threshold which determines if the gradient is positive or not.

Inputs

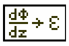
Name	Type	Default
Phase	Double Array	
Threshold	Double	0.8
Unwrapping Type	Enumeration	

Outputs

Name	Type
Unwrapped Phase	Double Array

C.6. Strain.llb

C.6.1 Calculate Strain Coefficient



This VI calculates the strain coefficient from several physical parameters of the fibre combined into a cluster.

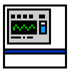
Inputs

Name	Type	Units
Fibre Parameters	**Cluster**	
Fibre Index	Double	
Unstrained Peak Wavelength	Double	m
Poisson's Ratio (v)	Double	
Strain-Optic Coefficient p11	Double	
Strain-Optic Coefficient p12	Double	

Outputs

Name	Type
Strain Coefficient	Double

C.6.2 Fibre Parameters



This VI displays a dialog box allowing the user to adjust the physical parameters of the fibre. The input and output values are clustered for use with the 'Calculate Strain Coefficient' VI.

Inputs

Name	Type
Input	Cluster

Outputs

Name	Type
Output	Cluster

C.6.3 Reverse Unwrapping Type

This simple VI reversed the direction of the unwrapping type for the 'Create Complex Reflectance' VI. Because of a negative sign in the formula, forcing the strain profile to be positive requires a negative gradient in the phase profile.
If the 'Threshold' setting is passed to this VI it is not changed.

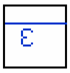
Inputs

Name	Type
Unwrapping Type	Enumeration

Outputs

Name	Type
Reversed Unwrapping Type	Enumeration

C.6.4 Strain Baseline Type

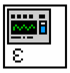


This VI displays a dialog box allowing the user to select the type of strain baseline desired. It takes no inputs. If the OK button is pressed then the OK output will be True.

Outputs

Name	Type
Selected Baseline Type	Enumeration
Selected Constant	Double
OK	Boolean

C.6.5 Strain User Interface



This VI controls the strain calculation interface. More information about strain calculation can be found in Appendix A.7. The Interface is divided into three sections. The final section, post-processing is disabled until the strain is calculated, unless values are passed to the Current Strain Position and Current Strain inputs.

Inputs

Name	Type	Units	Default
Input Reflectivity Wavelength	Double Array	m	
Input Reflectivity	Double Array		
Input Phase Wavelength	Double Array	m	
Input Phase	Double Array	rad	
Current Strain Position	Double Array	m	Empty
Current Strain	Double Array		Empty

Outputs

Name	Type	Units
Averaged Strain Position	Double Array	m
Averaged Strain	Double Array	

Page classification: UNCLASSIFIED

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION DOCUMENT CONTROL DATA	1. PRIVACY MARKING/CAVEAT (OF DOCUMENT)

2. TITLE	3. SECURITY CLASSIFICATION (FOR UNCLASSIFIED REPORTS THAT ARE LIMITED RELEASE USE (L) NEXT TO DOCUMENT CLASSIFICATION)
Operation Manual for Measurement and Discrete Layer Peeling of Fibre Bragg Grating Spectra	Document (U) Title (U) Abstract (U)
4. AUTHOR(S)	5. CORPORATE AUTHOR
Chris Brooks and Claire Davis	DSTO Defence Science and Technology Organisation 506 Lorimer St Fishermans Bend Victoria 3207 Australia

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO-TN-0801	AR-014-092	Technical Note	February 2008

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. NO. OF PAGES	12. NO. OF REFERENCES
2007/1105302/1	LRR06/063	DCDS(PHS)	60	0

13. URL on the World Wide Web	14. RELEASE AUTHORITY
http://www.dsto.defence.gov.au/corporate/reports/DSTO-TN-0801.pdf	Chief, Air Vehicles Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for public release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, EDINBURGH, SA 5111

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CITATION IN OTHER DOCUMENTS Yes

18. DSTO RESEARCH LIBRARY THESAURUS http://web-vic.dsto.defence.gov.au/workareas/library/resources/dsto_thesaurus.htm

Fibre Optic Sensors, Bragg Gratings, Distributed Sensors, Strain Measurement, User Manual

19. ABSTRACT

This document is a manual detailing the operation of an interrogation system for measuring the complex reflection spectrum of a Bragg grating. The manual also describes custom-designed software for deconvolving this data to determine the pitch profile of the grating by using a discrete layer peeling technique. This information can be used to determine the strain profile experienced by the grating in cases where the grating length spans the localised strain field.

Page classification: UNCLASSIFIED